



Universidade do Minho
Escola de Engenharia
Mestrado Integrado em Engenharia Informática

Unidade Curricular de Laboratórios de Informática IV

Ano Letivo de 2018/2019

Bela Sopa

Alberto Campinho Faria (A79077)
André Guilherme Nunes Viveiros (A80524)
César Augusto da Costa Borges (A81644)
Fábio Rafael Correia Guerra Fontes (A78650)
Luís José Rodrigues da Silva Macedo (A80494)

Junho, 2019

LI4

Data de Recepção	
Responsável	
Avaliação	
Observações	

Bela Sopa

Alberto Campinho Faria (A79077)

André Guilherme Nunes Viveiros (A80524)

César Augusto da Costa Borges (A81644)

Fábio Rafael Correia Guerra Fontes (A78650)

Luís José Rodrigues da Silva Macedo (A80494)

Junho, 2019

Resumo

Este documento consiste no relatório correspondente ao trabalho prático realizado no âmbito da Unidade Curricular de Laboratórios de Informática IV, do curso de Mestrado Integrado em Engenharia Informática da Universidade do Minho, no ano letivo de 2018/2019. Considera-se o caso de estudo do desenvolvimento do assistente pessoal de cozinha *Bela Sopa*, a pedido da cadeia de supermercados e hipermercados *Gota Doce*.

É primeiramente fundamentada a construção do sistema tendo em conta a sua utilidade e viabilidade, elaborando-se também um plano para o seu desenvolvimento. O processo de especificação do sistema é em seguida pormenorizado, apresentando-se com particular detalhe os resultados das fases de modelação de domínio, levantamento e análise de requisitos, modelação de *use cases*, prototipagem da interface de utilizador e modelação da arquitetura interna do sistema. É por fim descrito o processo de construção do mesmo e apresentado o produto resultante.

Área de Aplicação: Desenvolvimento de sistemas de software.

Palavras-Chave: Culinária; Assistente pessoal de cozinha; Sistemas de software; Engenharia de software.

Índice

1. Introdução	1
1.1. Contextualização e caso de estudo	1
1.2. Motivação e objetivos	2
1.3. Estrutura do relatório	2
2. Fundamentação do Sistema	4
2.1. Identidade do sistema	4
2.2. Justificação, viabilidade e utilidade do sistema	5
3. Planeamento do Projeto	6
3.1. Maqueta do sistema	6
3.2. Recursos necessários	7
3.3. Medidas de sucesso	7
3.4. Plano de desenvolvimento	8
4. Modelação de Domínio	11
5. Levantamento e Análise de Requisitos	13
5.1. Método adotado	13
5.2. Requisitos levantados	14
5.2.1. Requisitos de utilizador	14
5.2.2. Requisitos de sistema	15
6. Modelação de Use Cases	18
6.1. Identificação dos <i>use cases</i>	18
6.2. Especificação dos <i>use cases</i>	18
7. Prototipagem da Interface de Utilizador	21
8. Arquitetura do Sistema	26
9. Especificação da Camada de Dados	28
9.1. Diagramas de classes de objetos de acesso a dados	28

9.2. Modelação conceitual da base de dados	30
9.3. Modelação lógica da base de dados	30
10. Especificação da Camada de Negócio	32
10.1. Diagramas de classes	32
10.2. Diagramas de sequência	33
11. Construção	36
11.1. Tecnologias utilizadas	36
11.2. Detalhes de implementação	36
11.3. Procedimento de instalação	37
11.4. Produto final	38
12. Conclusões e Trabalho Futuro	43
Referências	44
Lista de Siglas e Acrónimos	45
Anexo I. Especificação dos Use Cases	46
Anexo II. Diagramas de Sequência para a Camada de Negócio	50

Índice de Figuras

1. Maqueta do sistema.	7
2. Diagrama de Gantt para o desenvolvimento do projeto.	8
3. Custos associados a cada fase do projeto.	9
4. Custos associados a cada contribuidor do projeto.	10
5. Diagrama do modelo de domínio.	12
6. Protótipo da interface de autenticação.	21
7. Protótipo da interface de criação de contas.	22
8. Protótipo geral das interfaces de cliente.	23
9. Protótipo da interface de listagem de receitas.	24
10. Protótipo da interface de visualização de uma receita.	24
11. Protótipo da interface de confeção de uma receita.	25
12. Diagrama de pacotes do sistema.	27
13. Diagrama de classes para a camada de dados.	29
14. Modelo conceitual da base de dados.	30
15. Modelo lógico da base de dados.	31
16. Diagrama de classes para a camada de negócio.	33
17. Diagrama de sequência para a confeção de uma receita.	34
18. Diagrama de sequência para a consulta de uma receita.	35
19. Aspetto final da interface de autenticação.	39
20. Aspetto final da interface de listagem de receitas.	40
21. Aspetto final da interface de listagem de ingredientes.	41
22. Aspetto final da interface de visualização de uma receita.	42
23. Diagrama de sequência para o acesso a serviços externos.	50
24. Diagrama de sequência para o adicionar de uma receita aos favoritos.	51
25. Diagrama de sequência para a remoção de uma conta.	51
26. Diagrama de sequência para a edição da ementa semanal.	52
27. Diagrama de sequência para a geração de uma lista de ingredientes.	53

28. Diagrama de sequência para o efetuar de um pedido de ajuda.	53
29. Diagrama de sequência para a procura de uma receita.	54
30. Diagrama de sequência para a criação de uma conta.	55
31. Diagrama de sequência para a visualização de lojas próximas.	55
32. Diagrama de sequência para a remoção de uma receita dos favoritos.	56

Índice de Tabelas

1. Ficha de projeto.	4
2. Especificação do <i>use case</i> “registrar conta”.	19
3. Especificação do <i>use case</i> “procurar receitas”.	19
4. Especificação do <i>use case</i> “confeccionar receita”.	20
5. Especificação do <i>use case</i> “aceder a serviços externos”.	46
6. Especificação do <i>use case</i> “adicionar receita aos favoritos”.	46
7. Especificação do <i>use case</i> “apagar conta”.	47
8. Especificação do <i>use case</i> “consultar ou editar informação da conta”.	47
9. Especificação do <i>use case</i> “consultar receita”.	47
10. Especificação do <i>use case</i> “editar ementa semanal”.	48
11. Especificação do <i>use case</i> “gerar lista de ingredientes”.	48
12. Especificação do <i>use case</i> “pedir ajuda”.	49
13. Especificação do <i>use case</i> “remover receita dos favoritos”.	49
14. Especificação do <i>use case</i> “ver loja mais próxima”.	49

1. Introdução

Este relatório documenta o trabalho desenvolvido no âmbito da Unidade Curricular de Laboratórios de Informática IV, do curso de Mestrado Integrado em Engenharia Informática da Universidade do Minho, no ano letivo de 2018/2019.

O caso de estudo considerado centra-se no desenvolvimento do assistente pessoal de cozinha *Bela Sopa*, sucessor ao serviço *online* intitulado *Escola de Cozinha* disponibilizado pela cadeia de supermercados e hipermercados portuguesa *Gota Doce*. Este capítulo contextualiza e apresenta o caso de estudo, descrevendo também as motivações e os objetivos do projeto, e delinea a estrutura do presente relatório.

1.1. Contextualização e caso de estudo

O *Gota Doce* é uma cadeia de supermercados e hipermercados portuguesa, sediada em Lisboa. Foi fundada em 1983 pelo grupo empresarial *Merónimo Jartins* em colaboração com a empresa belga *Gelhaize Group*. A cadeia possui atualmente mais de 400 lojas físicas distribuídas por cerca de 290 localidades portuguesas, contando com aproximadamente 32 milhares de colaboradores e 700 mil visitas diárias.

Além desta vasta rede de pontos de serviço, a empresa *Gota Doce* oferece também vários serviços *online*, destacando-se o serviço de entrega de produtos ao domicílio. Outros serviços têm como objetivo a fidelização de clientes e a manutenção da imagem da marca, como é o caso da *Escola de Cozinha*, uma plataforma *online* que disponibiliza informação relacionada com culinária. Esta oferece auxílio à confeção de refeições por parte dos seus utilizadores, fornecendo principalmente descrições estáticas de receitas, técnicas de cozinha e ingredientes.

Sendo este serviço utilizado por uma quantidade considerável de clientes da cadeia *Gota Doce*, a empresa pretende agora investir na melhoria da plataforma, focando a possibilidade de adaptação e personalização do serviço a cada um dos seus utilizadores. Com esse objetivo, foi pedido o desenvolvimento de um assistente pessoal de cozinha que substituirá o atual serviço *Escola de*

Cozinha. Por forma a se provar a viabilidade e adequação desta nova plataforma, a sua versão inicial será dedicada exclusivamente à confeção de sopas. O caso de estudo no qual o trabalho documentado neste relatório se baseia corresponde então ao desenvolvimento do assistente em causa, intitulado *Bela Sopa*.

1.2. Motivação e objetivos

Embora não se pretenda monetizar diretamente a plataforma em questão (e.g., através de subscrições pagas para acesso aos serviços por esta disponibilizados), objetiva-se com a sua construção angariar e fidelizar clientes para os principais serviços oferecidos pela empresa *Gota Doce*. A título de exemplo, o sistema poderá promover a utilização desses outros serviços ao indicar que os ingredientes utilizados por uma determinada receita podem ser obtidos em lojas físicas *Gota Doce* próximas ou através do serviço de entrega ao domicílio disponibilizado pela empresa. A plataforma poderá também aumentar a exposição dos clientes da cadeia a folhetos promocionais e outros materiais publicitários.

De forma geral e sumária, pretende-se que a construção da plataforma em questão contribua para o crescimento e manutenção do volume de negócio da empresa *Gota Doce*. Estes objetivos serão detalhados e quantificados em secção posterior deste relatório.

1.3. Estrutura do relatório

Este documento apresenta a seguinte estrutura:

- No Capítulo 2 fundamenta-se o sistema, tendo em conta a sua utilidade e viabilidade e justificando-se o seu desenvolvimento;
- No Capítulo 3 é apresentado o planeamento do projeto, juntamente com um modelo inicial do sistema, recursos necessários à concretização do mesmo, medidas de sucesso e um plano detalhado do seu desenvolvimento;
- No Capítulo 4 inicia-se a fase de especificação do sistema descrevendo-se o processo de modelação de domínio e os respetivos resultados;
- No Capítulo 5 apresenta-se o levantamento e análise de requisitos, enumerando-se os requisitos identificados e efetuando-se uma análise geral dos mesmos;
- No Capítulo 6 procede-se à modelação de *use cases*, começando-se por identificar a totalidade dos *use cases* considerados e apresentando-se depois a sua especificação;

- No Capítulo 7 é executada a prototipagem da interface de utilizador do sistema, recorrendo-se primariamente a esquemas do seu aspeto gráfico;
- No Capítulo 8 apresenta-se e especifica-se de forma holística a arquitetura interna do sistema;
- No Capítulo 9 detalha-se a especificação da camada de dados do sistema, modelando-se também a base de dados subjacente;
- No Capítulo 10 especifica-se da camada de negócio do sistema recorrendo primariamente a diagramas de classes e de sequência;
- No Capítulo 11 descreve-se a fase de construção do sistema, apresentando-se também o produto resultante;
- No Capítulo 12 o relatório é concluído com um resumo do processo de desenvolvimento do sistema e com a identificação de trabalho futuro.

Adicionalmente, são incluídos dois anexos:

- No Anexo I são reproduzidas as especificações de todos os *use cases* identificados no Capítulo 6;
- No Anexo II são incluídos todos os diagramas de sequência relativos à camada de negócio descrita no Capítulo 10.

2. Fundamentação do Sistema

Tendo-se apresentado o caso de estudo e identificado as motivações e objetivos para o desenvolvimento do sistema em questão, este capítulo define agora a sua identidade e fundamenta a sua construção tendo em conta a viabilidade e utilidade do mesmo.

2.1. Identidade do sistema

O *Bela Sopa* consiste num assistente pessoal que tem como objetivo acompanhar o utilizador durante todo o processo de confeção de uma receita de sopa, destinando-se à faixa etária adulta e não sendo direcionado a menores de idade.

Durante a confeção da receita, o assistente é capaz de interagir com o utilizador, designando quais os ingredientes necessários e os passos serem a executados, proporcionando também informação relativa aos utensílios e técnicas a utilizar durante a produção do prato, sendo assim capaz de responder a diferentes cenários alternativos.

Várias características que definem a identidade do sistema são apresentadas na Tabela 1.

Nome	<i>Bela Sopa</i>
Categoria	Assistente pessoal
Designação	Assistente pessoal de cozinha para a confeção de sopas
Idioma	Português
Faixa etária	Adultos
Características	<i>User friendly</i> , personalizável e prático
Empresa cliente	<i>Gota Doce</i>

Tabela 1: Ficha de projeto.

2.2. Justificação, viabilidade e utilidade do sistema

Justifica-se a realização deste projeto para substituir o serviço *Escola de Cozinha* por um sistema interativo, eficiente, inteligente e de melhor qualidade. Com este projeto a nossa empresa obtém um cliente de grande escala enquanto que o cliente obtém um serviço inovador.

Antes do nosso cliente ter investido neste projeto, teve que ter um contexto para o mesmo. Para isso, este teve de investigar os seus clientes atuais e desejados para entender a vida dos mesmos e as suas necessidades, ficando com uma ideia do futuro do projeto para atender às demandas dos seus clientes.

Isto foi feito através de um questionário, o qual se concluiu que cerca de 80% das pessoas, tanto clientes atuais como possíveis futuros clientes, procuram respostas tecnológicas aos seus problemas diários, e que 70% destes utilizariam um assistente pessoal para ajudar na sua culinária, enquanto que apenas 8% possuem elevada experiência culinária.

Este projeto é viável pelas seguintes razões:

- Não existem obstáculos legais (leis ou patentes) que proíbam ou limitem o desenvolvimento e comercialização do produto resultante;
- A nossa equipa prevê que o desenvolvimento deste produto é exequível com os recursos tecnológicos atuais;
- É geograficamente favorável devido ao elevado número de super e hipermercados e à sua dispersão pelo território nacional;
- É vantajoso em termos de visibilidade pública, uma vez que a cadeia *Gota Doce* está associada ao produto;
- Não existe impactos ambientais associados ao produto;
- Financeiramente, prevê-se que o projeto seja rentável a curto prazo.

3. Planeamento do Projeto

Estando fundamentada a construção do assistente *Bela Sopa*, apresenta-se agora o planeamento do seu desenvolvimento. Começa-se por se definir um modelo inicial do sistema, procedendo-se com a identificação de recursos necessários à sua concretização e de medidas de sucesso. Por fim, apresenta-se um plano cronológico detalhado do seu desenvolvimento.

3.1. Maqueta do sistema

Embora não seja ainda possível determinar o conjunto exato de funcionalidades desejadas e a estrutura interna do sistema a ser desenvolvido, podem já ser identificados os seus principais tipos de utilizador e componentes, assim como possíveis dependências em serviços externos.

Com base nessa informação, foi elaborado um modelo inicial do sistema (ou *maqueta*), o qual é apresentado na Figura 1. Identificam-se dois tipos de utilizador: *administradores* — responsáveis pela gestão dos conteúdos disponibilizados — e *clientes* — correspondentes ao público-alvo do sistema *Bela Sopa*. Na figura, a entidade ou componente na origem de cada seta inicia interações com o componente no destino da mesma.

Prevê-se a possibilidade de a plataforma fazer uso de um sistema de informação detido pela empresa *Gota Doce* por forma a obter dados sobre produtos e lojas da mesma e a tirar partido da base de informação relativa a receitas culinárias e ingredientes atualmente em uso pelo serviço *Escola de Cozinha*. No entanto, prevê-se também que a plataforma necessite de uma base de informação própria, dedicada à gestão de dados específicos ao serviço a ser desenvolvido.

Explicita-se que este modelo é provisório e não vinculativo, tendo o principal objetivo de auxiliar a identificação de recursos necessários ao desenvolvimento do sistema e o início da fase de especificação do mesmo.

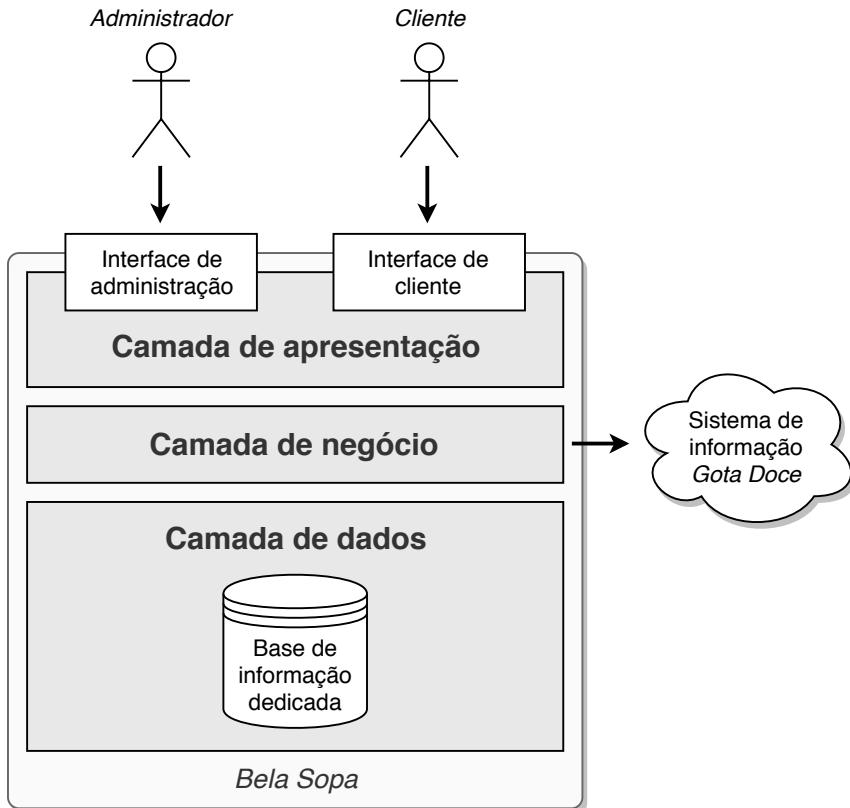


Figura 1: Maqueta do sistema.

3.2. Recursos necessários

Estabeleceu-se que, para o desenvolvimento do sistema, é necessária uma equipa de desenvolvimento de 8 elementos, respetivamente, 1 gestor, 1 analista, 4 programadores e 2 engenheiros de software.

Para além disso, será necessário utilizar os recursos disponibilizados pela empresa *Gota Doce*, como a base de receitas da plataforma *Escola de Cozinha*, por forma a obter o nome das receitas, a dificuldade, a duração, os ingredientes, os passos da receita e as porções concebidas para as sopas, e um consultor que nos dará a informação necessária sobre culinária.

3.3. Medidas de sucesso

As medidas de sucesso aqui enumeradas foram estabelecidas através de uma reunião entre o nosso gestor e a equipa de contabilidade do nosso cliente, sendo que os dados referidos foram especulados com base em 6 meses de uso do sistema após o seu lançamento.

Estima-se que:

- A venda de produtos aumente em, pelo menos, 5% através da utilização do sistema;
- O sistema seja utilizado em média por 1000 utilizadores distintos por dia;
- O número de clientes da cadeia aumente em, pelo menos, 3%;
- O número de pedidos efetuados através do serviço de entrega ao domicílio aumente em, pelo menos, 10%.

3.4. Plano de desenvolvimento

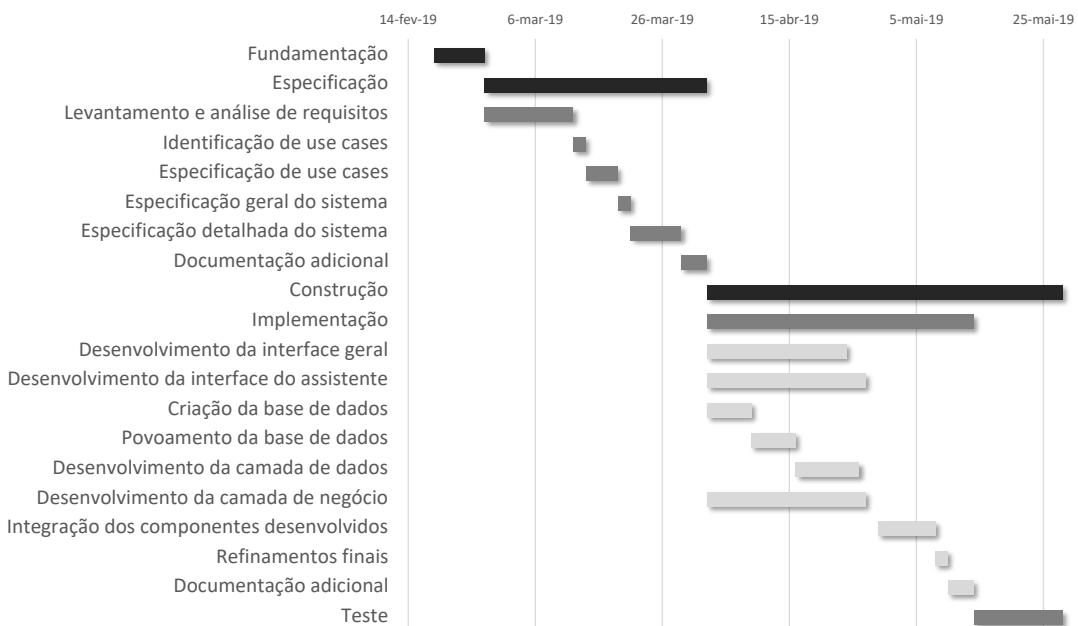


Figura 2: Diagrama de Gantt para o desenvolvimento do projeto.

A análise e fundamentação do projeto será feita pelo analista da equipa, e os restantes processos só serão feitos após esta fase, com uma duração de 6 dias.

Durante o desenvolvimento do sistema estimam-se 6 tarefas necessárias, respetivamente, levantamento de requisitos, análise de requisitos, arquitetura de software, testes e implementação. O gestor e os colaboradores correspondentes à fase atual do projeto, irão se reunir duas vezes por semana de forma a distribuir tarefas para permitir trabalho autónomo e debater eventuais dúvidas, conflitos ou problemas no trabalho a realizar.

O levantamento de requisitos e sua respetiva análise terá a duração de 3 dias e será realizado pelos engenheiros de software e com o auxilio de um consultor ainda a ser disponibilizado pelo

Gota Doce.

A identificação e especificação de use cases demorará 4 dias e será feito após o levantamento e análise de requisitos. A arquitetura de software terá um espaço de 4 semanas e será realizada pelos engenheiros de software. A implementação terá a duração de 6 semanas e será realizada pelos programadores.

Os testes serão feitos durante 10 dias por um programador da equipa de desenvolvimento e um outro programador que desconhece o produto, de forma a que os testes sejam feitos de forma imparcial e independentes do facto do colaborador ter desenvolvido o produto ou não. Após isto, durante 4 dias será realizada a apresentação e instalação do sistema de software nos ambientes do cliente.

Em suma, estimam-se 14 semanas para o desenvolvimento deste sistema.

O custo associado a cada uma das fases do projeto encontra-se na Figura 3, estando associado ao custo inicial a fase "Análise e fundamentação". Em suma o custo total do projeto é de 30.052,00 euros.

Nome	Custo Fixo	Custo Real	Custo Restante	Custo	Custo do Plano Base	Desvio de Custos
Análise e fundamentação	0,00 €	528,00 €	0,00 €	528,00 €	0,00 €	528,00 €
Especificação	0,00 €	0,00 €	11 044,00 €	11 044,00 €	0,00 €	11 044,00 €
Construção	0,00 €	0,00 €	18 480,00 €	18 480,00 €	0,00 €	18 480,00 €

Figura 3: Custos associados a cada fase do projeto.

Estes custos foram gerados tendo em conta 8 horas de trabalho por dia, 5 dias por semana, e os custos para cada elemento individualmente que se encontram na Figura 4.

Nome	Trabalho Real	Custo Real	Taxa Padrão
Fábio	0 hrs	0,00 €	11,00 €/hr
Mr.Burns	0 hrs	0,00 €	27,50 €/hr
André	0 hrs	0,00 €	16,50 €/hr
Guilherme	0 hrs	0,00 €	11,00 €/hr
César	0 hrs	0,00 €	16,50 €/hr
Alberto	0 hrs	0,00 €	11,00 €/hr
Luís	0 hrs	0,00 €	11,00 €/hr
Augusto	48 hrs	528,00 €	11,00 €/hr

Figura 4: Custos associados a cada contribuidor do projeto.

4. Modelação de Domínio

Verificou-se que a construção do sistema seria vantajosa para a empresa e que o seu processo de desenvolvimento cumpriria o orçamento e prazos estabelecidos, reunindo-se assim as condições necessárias para se proceder à fase de especificação do mesmo.

Embora os procedimentos de modelação de domínio e de levantamento de requisitos sejam mutuamente dependentes, apresenta-se em primeiro lugar neste relatório o modelo de domínio por forma a clarificar a enunciação dos requisitos apresentados no próximo capítulo.

A abordagem utilizada para efetuar a modelação é baseada naquela apresentada por Sommerville (2010), sendo utilizada a notação gráfica *Unified Modeling Language* (UML) (Object Management Group, 2017), por representar a linguagem padrão para modelação orientada a objetos.

A modelação de domínio foi efetuada em conjunto com o cliente do projeto, para garantir a correta estruturação do sistema, uma adequada nomenclatura que contribua para uma melhor percepção do sistema e que o sistema é apropriado para o seu objetivo.

Decidiu-se que o sistema seria centrado na receita, que terá sempre uma proporção, uma dificuldade, vários ingredientes, técnicas e/ou utensílios de cozinha envolvidos, um conjunto de processos, cada um com um conjunto de tarefas que devem ser realizadas antes de prosseguir para o próximo processo, estando um tempo de execução e designação associada a cada tarefa e receita. A receita também poderá ter valores nutricionais e etiquetas associadas a esta.

Quanto ao utilizador, este terá um nome, um email, uma palavra-passe e um distrito associado à sua conta pessoal, podendo este ser um cliente ou um administrador. Um cliente terá acesso a uma ementa semanal, com as receitas que pretende confeccionar, para o almoço e jantar, nos dias de uma semana, e uma localização geográfica.

A localização geográfica é também atribuída à loja onde o cliente poderá encontrar os diversos ingredientes usados nas receitas.

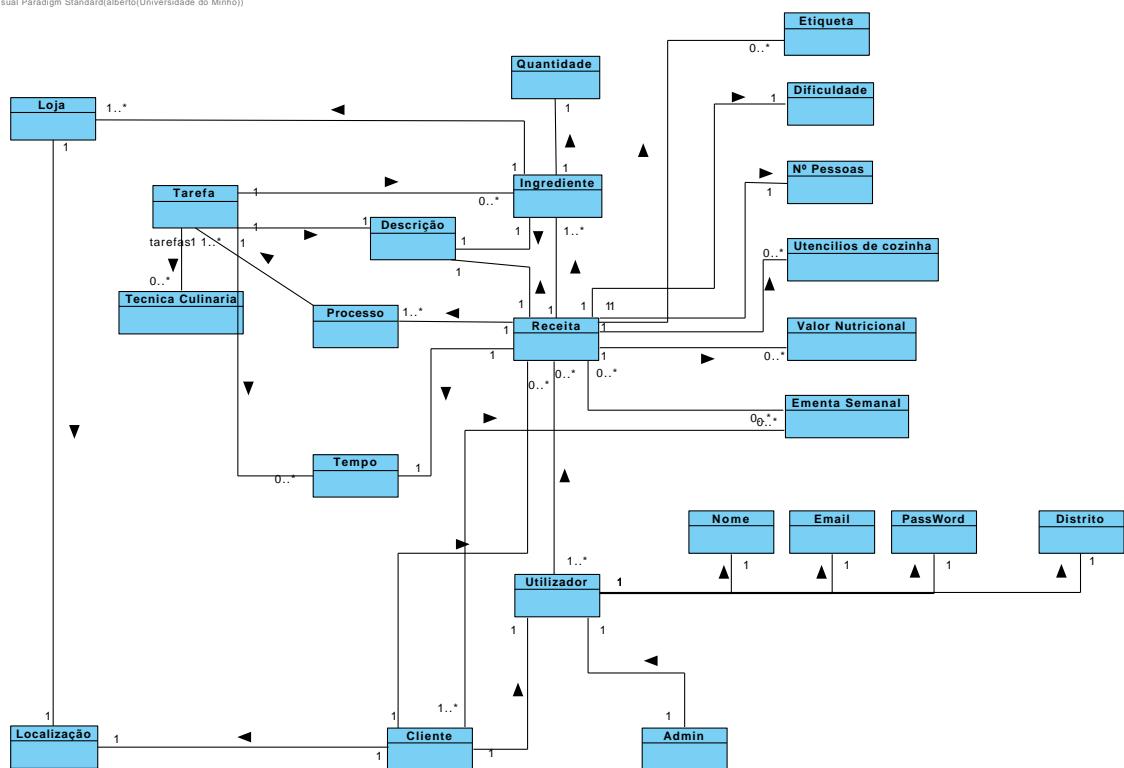


Figura 5: Diagrama do modelo de domínio.

5. Levantamento e Análise de Requisitos

Tendo-se efetuado a modelação de domínio, procedeu-se ao levantamento e análise de requisitos. Neste capítulo é primeiramente descrito o método utilizado para se efetuar a identificação de requisitos, enumerando-se de seguida os requisitos identificados. Estes utilizam o glossário definido pelo modelo conceitual apresentado no capítulo anterior.

5.1. Método adotado

Durante a recolha de requisitos, recorreu-se a várias estratégias por forma a garantir que todos os requisitos necessários eram registados.

Uma vez que a nossa percepção do sistema é afetada pelas nossas experiências e gostos não é suficiente entender o que se pretende que o sistema faça, mas também adaptar o sistema aos gostos dos utilizadores por forma a encontrar um ponto de consenso e fornecer a melhor experiência possível a todos os utilizadores.

Assim para além do cliente da aplicação, o foco da nossa atenção na aquisição de requisitos são os atuais utilizadores do sistema *Escola de Cozinha* e outros que constituem potenciais utilizadores, para que os requisitos sejam o mais abrangentes possíveis.

Foi assim proposto um inquérito aos utilizadores do sistema *Escola de Cozinha*, para que estes partilhassem que funcionalidades extra consideravam necessárias no sistema, e analisados os dados dessa primeira plataforma de forma a obter os hábitos dos utilizadores e podermos garantir a presença e melhoria das funcionalidades mais utilizadas.

Para obter a opinião de potenciais utilizadores recorremos à partilha de questionários em plataformas de comunicação, bem como questionários presenciais em algumas das lojas da empresa *Gota Doce*.

Assim da conjugação de todas as informações obtidas são extraídos os requisitos, que são posteriormente validados pela empresa cliente *Gota Doce*.

5.2. Requisitos levantados

Os requisitos são divididos em *requisitos de utilizador* e *requisitos de sistema*, de acordo com a distinção definida por Sommerville (2010, Capítulo 4), para facultar uma exposição clara. São também apresentados requisitos de sistema não-funcionais.

5.2.1. Requisitos de utilizador

Tipos de utilizador, contas de utilizador e autenticação.

- U.1. O sistema deverá suportar dois tipos de utilizador: (1) *clientes* — o público-alvo do sistema — e (2) *administradores* — utilizadores responsáveis pela gestão da informação disponibilizada pelo sistema.
- U.2. Ao administrador é permitido a adição ou remoção de receitas do sistema.
- U.3. O utilizador pode alterar/remover informações da sua conta.

Requisitos de execução da confeção.

- U.4. No inicio da confeção, o utilizador pode confirmar que possui todos os ingredientes para a receita.
- U.5. O utilizador pode recomeçar um processo quando quiser, pode ainda voltar para um processo anterior ou avançar para o próximo, caso exista.
- U.6. Em qualquer processo da confeção da receita, em caso de dúvida o utilizador pode pedir informações ao sistema, tais como: (1) o utilizador pode pedir informação extra das técnicas de utensílio e/ou materiais no processo específico; (2) clarificar certos termos, mostrando sinónimos a um termo; (3) quanto tempo demora o processo.
- U.7. Durante a confeção da sopa, em caso de uma tarefa com tempo de espera, este pode ativar um temporizador do sistema.
- U.8. Após a confeção da receita, o utilizador pode classificar a dificuldade que teve na execução da mesma.
- U.9. O utilizador pode cancelar a confeção em qualquer parte da realização da mesma.

Outros.

- U.10. O utilizador pode questionar o sistema relativamente às sua funcionalidades.
- U.11. O utilizador deve conseguir procurar receitas, técnicas e utensílios através do motor de busca (permitindo um acesso rápido ao que o utilizador deseja)
- U.12. Quando requerido, o sistema deve anotar uma ementa semanal e preparar a lista dos

- ingredientes necessários para cada refeição, gerando quando necessário uma lista de compras geral para a semana
- U.13. Visualizar num conjunto de *dashboards* (painéis de controlo) específico a um conjunto de dados relativos aos cozinhados realizados, os tempos de preparação, as dificuldades encontradas, os ingredientes utilizados, etc..
 - U.14. O utilizador pode definir uma configuração inicial para o sistema, deixando-os escolher aquilo que usualmente pretendem cozinhar e o tipo de ingredientes que querem usar.
 - U.15. O utilizador pode adicionar ou remover uma receita da sua lista de favoritos.
 - U.16. O utilizador pode definir uma configuração inicial para o sistema, indicando que tipo de receita pretende, usualmente, confeccionar e o quais os ingredientes que intendem utilizar.

5.2.2. Requisitos de sistema

Tipos de utilizador, contas de utilizador e autenticação.

- SF.1.1. A cada utilizador é atribuída apenas uma *conta de utilizador*.
 - SF.1.2. Cada conta de utilizador é classificada como *de administrador* ou *de cliente*, correspondendo ao tipo de utilizador homónimo, sendo identificada por um nome de utilizador e salvaguardada por meio de uma palavra-chave.
 - SF.1.3. Para utilizar o sistema, um utilizador deverá primeiro autenticar-se no mesmo indicando o nome de utilizador e a palavra-chave da sua conta.
 - SF.1.4. Deve sempre existir, no mínimo, uma conta de administrador.
 - SF.1.5. Um utilizador não autenticado pode criar contas de cliente, fornecendo um nome de utilizador, uma palavra-chave e, opcionalmente, um email.
 - SF.1.6. Um utilizador autenticado pode revogar a sua autenticação.
 - SF.1.7. Um administrador autenticado pode criar e eliminar contas de administrador.
 - SF.1.8. Um administrador autenticado pode eliminar contas de cliente.
 - SF.1.9. Um cliente autenticado pode eliminar a sua própria conta.
- SF.2.1. O administrador pode adicionar uma receita nova ao sistema, a qualquer momento, fornecendo todos os elementos constituintes da mesma.
 - SF.2.2. O administrador pode remover qualquer receita existente no sistema, bastando para tal especificar o identificador da mesma.
 - SF.2.3. Não é permitida a existência de duas receitas com o mesmo identificador
- SF.3.1. Toda a informação armazenada em associação à conta de um utilizador, pode ser modificada pelo mesmo, excetuando informações que seja utilizadas para efeitos de

identificação da conta.

- SF.3.2. Quaisquer informações opcionais, introduzidas pelo utilizador, podem ser removidas pelo mesmo

Requisitos de execução da confeção.

SF.4.1. O sistema deverá apresentar, em cada receita, uma breve descrição, imagens, número de doses, quais os utensílios e ingredientes necessários e as suas respetivas quantidades, passos da receita, tempo de confeção, dificuldade, etiquetas e uma breve informação nutricional.

SF.4.2. Caso o utilizador não tenha todos os ingredientes, este pode ser redirecionado para um serviço em que seja possível a compra / encomenda dos ingredientes.

SF.5.1. Cada processo deve possuir uma ou mais tarefas que são independentes entre elas.

SF.5.2. Só é permitido avançar para o próximo processo quanto todas as tarefas, do processo atual, forem concluídas, caso não haja mais processos a execução da receita termina.

SF.5.3. Sistema deve ser capaz de recomeçar a explicação do processo atual, ou voltar a explicar um processo prévio caso seja requerido.

SF.6.1. O sistema deve apresentar e explicar a receita e, caso o utilizador não saiba como concluir alguma tarefa, sugerir alguns vídeos de ajuda ou *websites* com informação pertinente.

SF.6.2. O sistema deve mostrar a lista de ingredientes, utensílios e técnicas de cozinha que são utilizados na execução da receita.

SF.6.3. As unidades utilizadas nos ingredientes são medidos com valores concretos e em alguns casos outras unidades equivalentes.

SF.6.4. O sistema pode mostrar a duração que cada tarefa tem, em média, e consequentemente, pode também mostrar a duração do processo.

SF.6.5. O sistema deve permitir a alteração das palavras técnicas por outras fornecidas pelo utilizador para simplificar os processos.

SF.6.6. Em fase de preparação, o sistema deve estar preparado para redirecionar o utilizador para uma das lojas mais próximas referentes ao mesmo com todos os ingredientes necessários para a confeção da sopa ou disponibilizar o serviço de entrega ao domicílio (O utilizador pode alterar estes ingredientes posteriormente).

SF.7.1. O sistema alerta o utilizador sonoramente quando o temporizador terminar.

SF.7.2. O temporizador deve ser apresentado em tempo real, mostrando os minutos e segundos restantes.

SF.8.1. A receita confeccionada é guardada num histórico.

SF.9.1. Ao cancelar a confeção é perdida qualquer informação relativa a esta.

Outros.

SF.10.1. O sistema deve apresentar os seus serviços e funcionalidades, explicando previamente o seu modo e princípios de funcionamento

SF.11.1. As receitas podem ser pesquisadas por nome ou etiqueta.

SF.11.2. Os utensílios podem ser pesquisados por nome.

SF.11.3. As técnicas podem ser pesquisadas por nome.

SF.12.1. A ementa semanal pode ser gerada na sua totalidade no seu menu adequado.

SF.12.2. No menu de uma receita, esta pode ser adicionada à ementa semanal nos dias e refeições escolhidas.

SF.12.3. Só se pode adicionar uma receita por refeição, 2 refeições por dia, 7 dias por semana

SF.14.1. É possível pré-definir o que se pretende cozinhar por etiquetas

SF.14.2. A configuração inicial pode ser editada mais tarde nas definições de conta

SF.15.1. As receitas encontradas pelos filtros usados na janela de pesquisa podem ser adicionadas da lista de favoritos, sendo possível a sua remoção da mesma.

SF.15.2. No final da confeção de uma receita, o utilizador pode adicioná-la à lista de favoritos, sendo possível, posteriormente, a sua remoção da lista.

Requisitos de sistema não-funcionais.

SNF.1. O sistema deverá disponibilizar uma interface de utilizador *web*, i.e., acessível através de um *web browser*.

SNF.2. O sistema deverá ser desenvolvido com recurso a tecnologias *Microsoft*.

SNF.3. Possuir um interface bastante amigável, suportado por um gestor de diálogos sofisticados e capaz de sustentar uma “conversa” razoável com o utilizador ao longo de todos os processos de trabalho, sendo capaz de especificar o passo em que se encontra e quais os passos realizados.

6. Modelação de *Use Cases*

Uma vez terminados os processos de modelação de domínio e de levantamento de requisitos, procedeu-se à modelação de *use cases*. Neste capítulo são apresentados os resultados dessa fase.

6.1. Identificação dos *use cases*

Os *use cases* foram identificados através da análise dos requisitos de utilizador, de forma a satisfazer todas as funcionalidades desejadas. Deste modo, os *use cases* identificados podem corresponder a uma ou mais funcionalidades, oferecidas pelo sistema, visto que, algumas delas correspondem a cenários alternativos de outras.

Foram subentendidos e omitidos no diagrama de *use cases*, *use cases* acessórios relativos à conta do utilizador, tais como, *logout* e *editar informações de conta*, por já possuírem um comportamento normalizado. Apenas foi mencionado o *use case Autenticar* pelo facto dos outros *use cases* possuírem como pré-condição a autenticação do utilizador, no entanto, não é apresentada a especificação deste.

6.2. Especificação dos *use cases*

Apresentam-se nesta secção alguns exemplos representativos de especificações de *use cases*. As especificações completas dos restantes *use cases* identificados são incluídas no Anexo I.

Use case:	Registrar conta	
Pré-condição:	Ator não existe	
Pós-condição:	Ator adicionado ao sistema	
	Autor	Sistema
Comportamento Normal		1. Pergunta o email, nome, distrito e palavra-chave
	2. Fornece dados	
		3. Valida dados
		4. Adiciona utente
Comportamento Alternativo 1 [Dados inválidos] (passo 3)		3.1. Indica ao cliente que os dados são inválidos.
		3.2. Volta para 1

Tabela 2: Especificação do *use case* “registrar conta”.

Use case:	Procurar receitas	
Pré-condição:	Estar autenticado	
Pós-condição:	Encontrou pelo menos uma receita	
	Autor	Sistema
Comportamento Normal	1. Introduz nome de uma receita	
		2. Inicia pesquisa da receita pelo nome
		3. Valida pesquisa
		4. Mostra receitas encontradas
Comportamento Alternativo 1 [Atr procura receitas por tag] (passo 1)	1.1. Introduz uma tag	
		1.2. Inicia pesquisa de receitas pela tag
		1.3. Volta ao passo 3
Exceção 1 [Nenhuma receita corresponde à pesquisa] (passo 3)		3.1. Indica que nenhuma receita foi encontrada

Tabela 3: Especificação do *use case* “procurar receitas”.

Use case:	Confeccionar receita	
Pré-condição:	Estar autenticado	
Pós-condição:	Receita confeccionada	
	Autor	Sistema
Comportamento Normal		1. Pergunta se possui todos os ingredientes.
	2. Confirma	
		3. Pergunta se possui todos os utensílios
	4. Confirma	
		5. Recebe o primeiro processo
		6. Mostra o processo atual, as tarefas neste e os ingredientes, utensílios e técnicas nas respectivas tarefas
	7. Confirma a execução de todas as tarefas	
	8. Avança	
		9. Verifica próximo processo
		10. Não existe mais processos
		11. Guarda a conclusão da receita
		12. Pergunta a opinião sobre a conceção da receita
	13. Responde	
		14. Guarda resposta
Comportamento Alternativo 1 [Existe próximo processo] (passo 10)		10.1. Recebe o próximo processo e volta ao passo 6
Comportamento Alternativo 2 [Utilizador retrocede o processo] (passo 7)		7.1. Recebe o processo anterior e volta ao passo 6
Comportamento Alternativo 3 [Processo anterior não existe] (passo 8)		8.1.1. Indica que não existe um processo anterior
		8.1.2. Mantém o processo atual
		8.1.3. Volta ao passo 6
Comportamento Alternativo 4 [Atores não respondem] (passo 13)		13.1. Termina Confeccionar receita
Exceção 1 [Não confirma] (passo 2 ou 4)		2.1. Indica para voltar quando possuir tudo
		2.2. Sai do serviço Confeccionar receita
Exceção 2 [Utilizador cancela Confeccionar receita] (passo indeterminado)		2.1. Sai do serviço Confeccionar receita

Tabela 4: Especificação do use case “confeccionar receita”.

7. Prototipagem da Interface de Utilizador

Antes de se proceder à modelação da arquitetura interna do sistema, construiu-se um protótipo da interface de utilizador disponibilizada pelo sistema, o qual é descrito neste capítulo.

Quando a aplicação é iniciada, são requisitas as credenciais, nome de utilizador e palavra-chave, para a autenticação do utilizador. É fornecida também a opção de criar uma conta pessoa. A criação de uma conta requer, para além das informações necessárias à autenticação, a confirmação da palavra-chave e opcionalmente a introdução de um endereço de *email*. Estas duas funcionalidades podem ser observadas na Figura 6 e Figura 7.

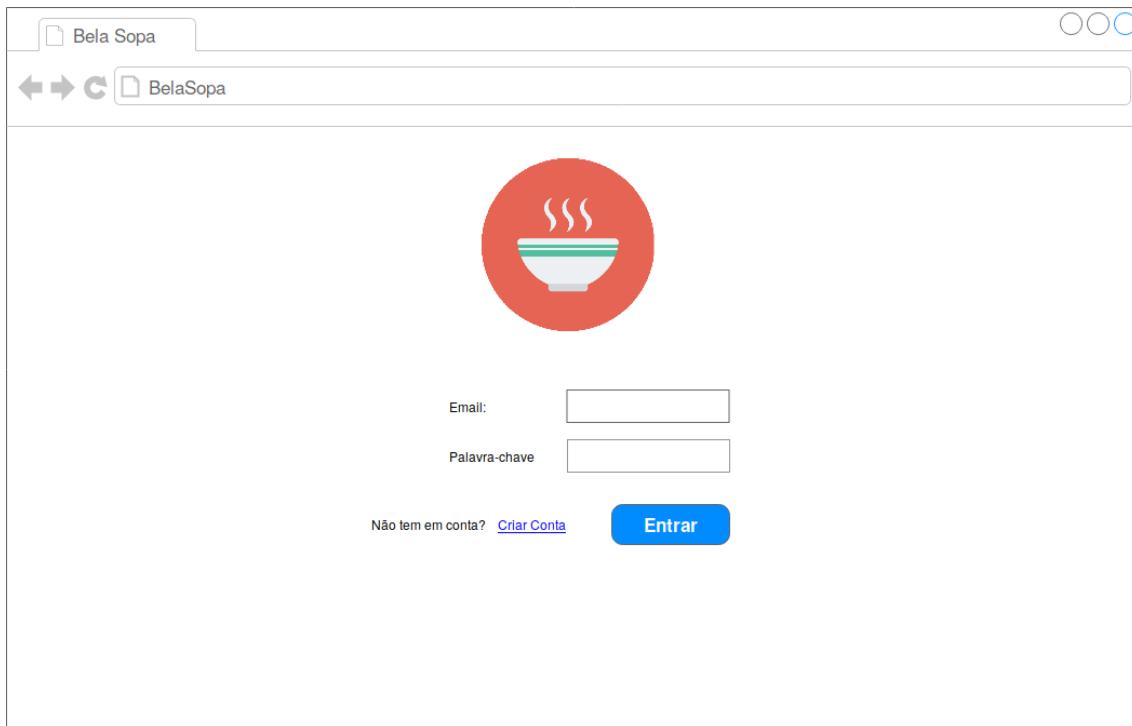


Figura 6: Protótipo da interface de autenticação.

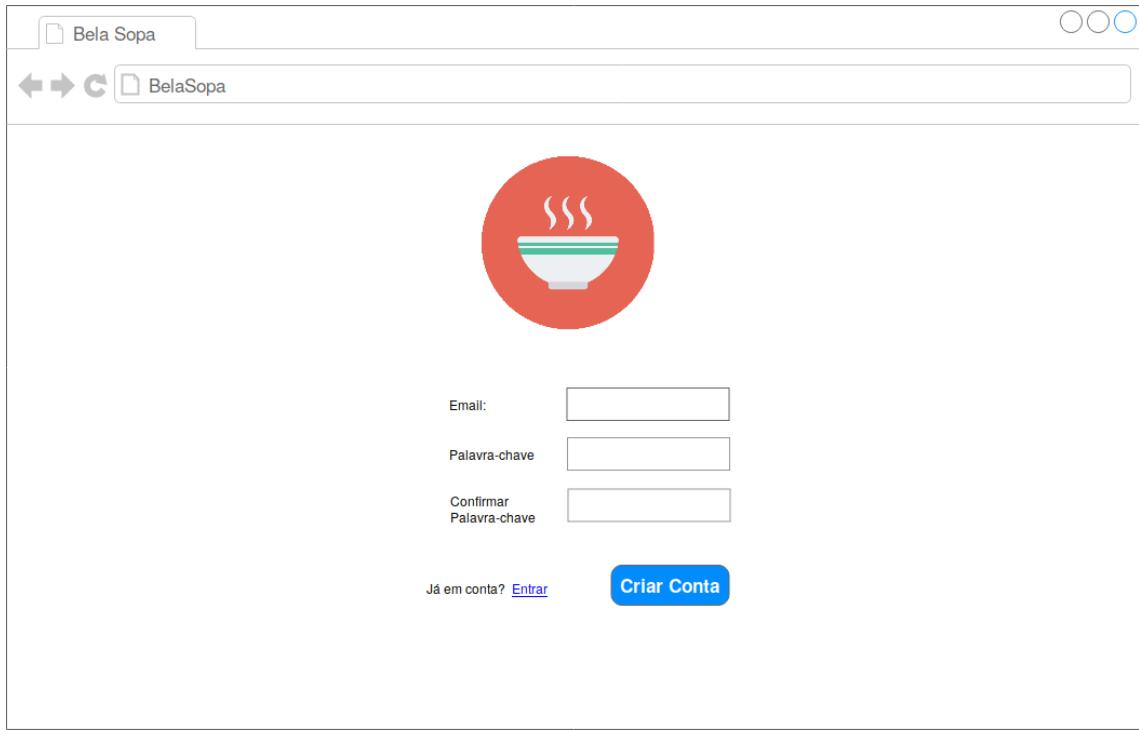


Figura 7: Protótipo da interface de criação de contas.

Após a autenticação, é apresentado ao utilizador uma lista com todas as receitas do sistema. Nessa mesma janela é visível um menu lateral onde o utilizador pode aceder:

- à sua conta pessoal;
- à receita que está em confeção (pode não estar visível);
- à lista de todas as receitas no sistema;
- à lista de todos os ingredientes no sistema;
- à lista de todas as técnicas no sistema;
- à lista de todos os ingredientes no sistema;
- à agenda semanal do utilizador;
- ao histórico do utilizador.
- à localização de lojas, *Gota Doce*, próximas.

, tal como se apresenta na Figura 8.



Figura 8: Protótipo geral das interfaces de cliente.

A janela de pesquisa, dá a liberdade ao utilizador de pesquisar uma receita por nome, etiqueta ou por dificuldade. Depois do filtro ser ativado, são apresentadas todas as receitas que são validas de acordo com o filtro. Na apresentação das receitas, é apresentado uma imagem, titulo e etiqueta de cada receita, tal como se demonstra na Figura 9

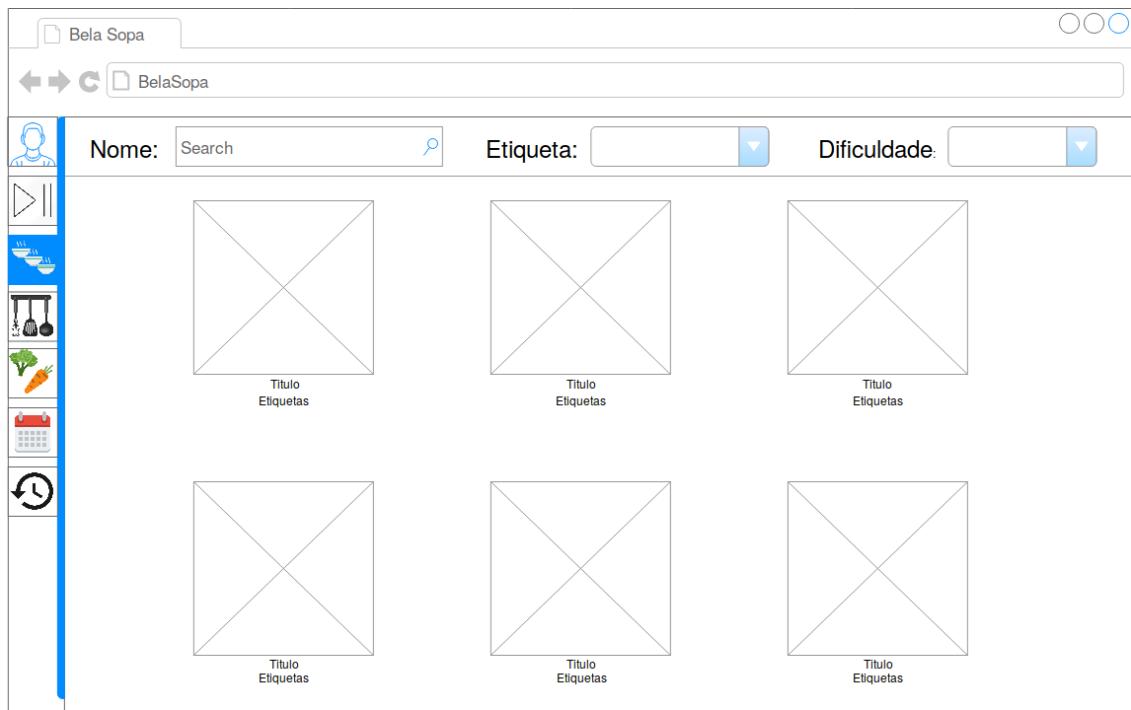


Figura 9: Protótipo da interface de listagem de receitas.

Quando o utilizador seleciona uma receita, esta é apresentada como na Figura 10, ou seja, é apresentado uma imagem da receita, o título, uma breve descrição, a dificuldade, a etiqueta, a duração, a porção, a lista de todos os ingredientes e as suas respetivas quantidades, a lista de utensílios, a lista de técnicas, a lista de tarefas e um conjunto de valores nutricionais.

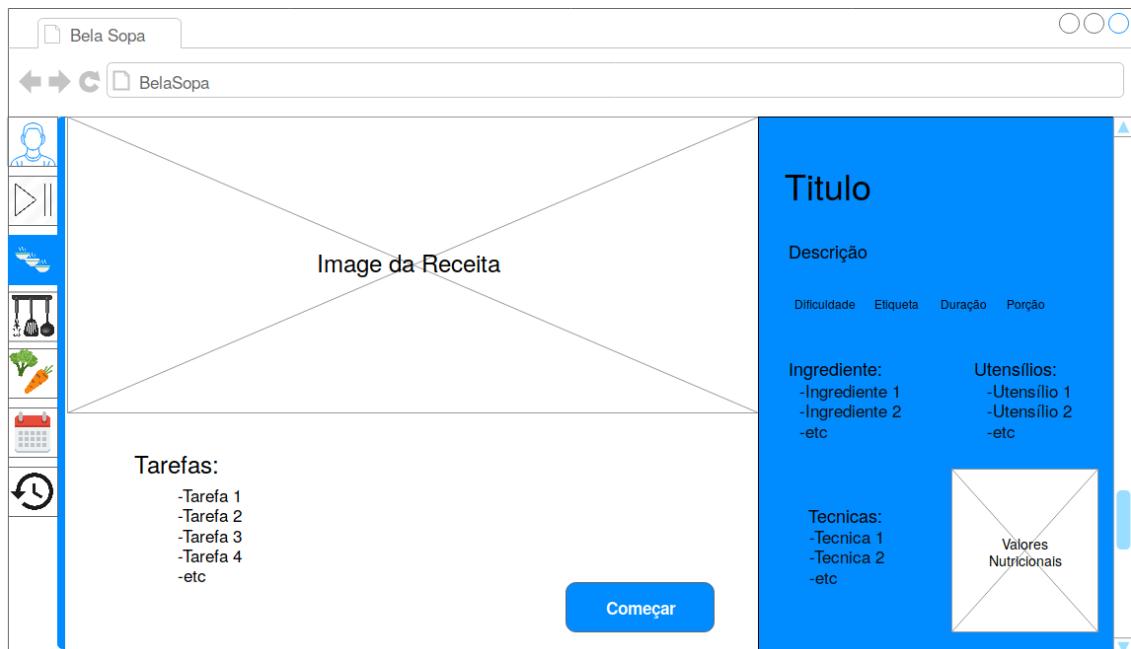


Figura 10: Protótipo da interface de visualização de uma receita.

Após iniciar a confeção, as informações da receita continuam a ser apresentados, exceto a lista de todos os ingredientes, de todos os utensílios e de todas as tarefas. Estas são reduzidas para cada processo, ou seja, em cada processo só são apresentadas as listas dos ingredientes, utensílios e tarefas referentes ao processo atual. Para além dessas informações, a janela permite ao utilizador avançar para o processo seguinte, recuar para o processo anterior, cancelar a confeção e iniciar um temporizador, tal como demonstrado na Figura 11.



Figura 11: Protótipo da interface de confeção de uma receita.

8. Arquitetura do Sistema

Por fim, procedeu-se à modelação da arquitetura interna do sistema. Neste capítulo estabelece-se a divisão do sistema em camadas, sendo que os próximos dois capítulos detalham esta modelação.

O sistema será dividido em três camadas principais:

1. a *View* que irá englobar tudo o que será necessário para a apresentação do sistema ao utilizador;
2. o *Model* que representa a camada de negócios do sistema, esta camada irá fornecer a informação necessária para a camada de apresentação;
3. a *Data* que representa os objetos de acesso à base de dados, estando cada um deles dependentes da existência do objeto respondente na camada de negócios.

A camada de negócios ainda é dividida em três sub-pacotes:

1. *Utilizador* que está relacionado com representação de todos os dados do utilizador;
2. *Loja* que representa os dados das lojas;
3. *Assistente* que está relacionado com o que os dados que o assistente manipulará na confecção das receitas

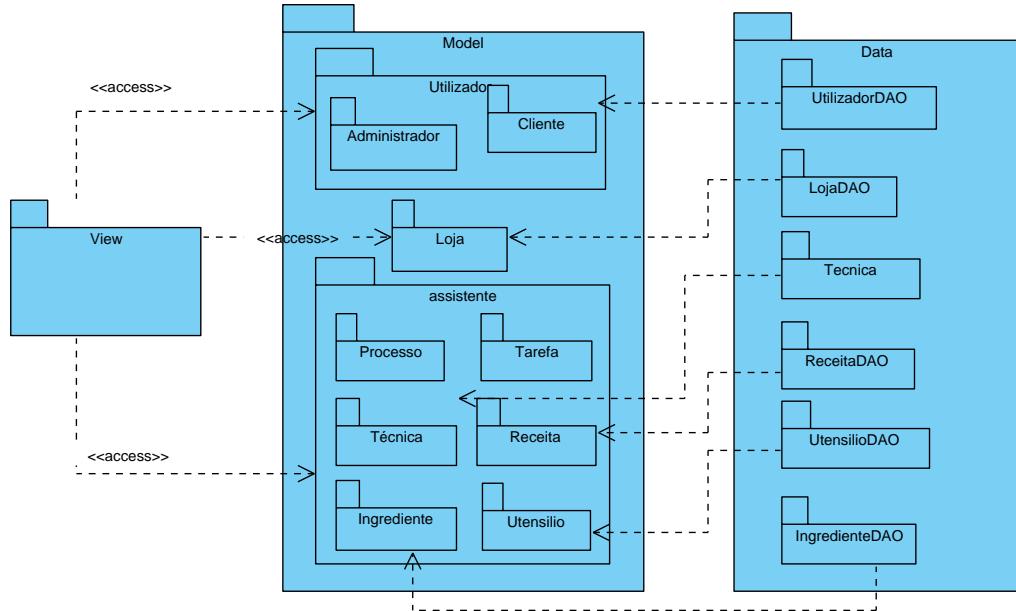


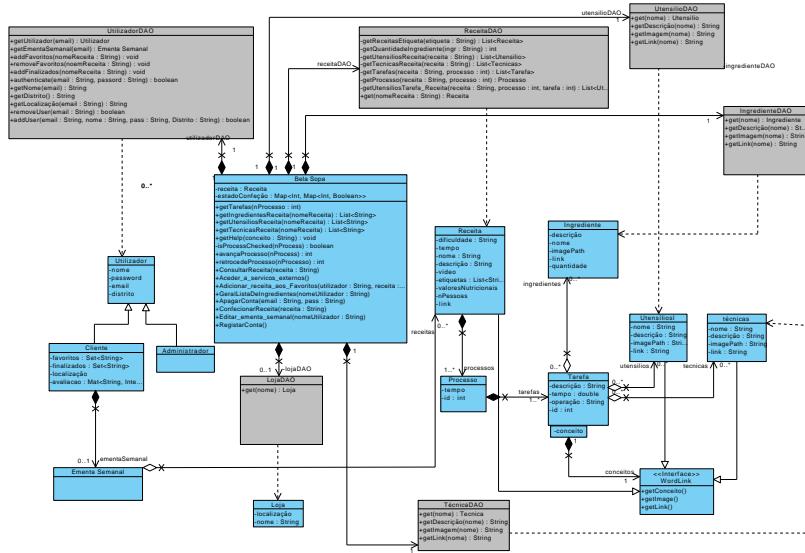
Figura 12: Diagrama de pacotes do sistema.

9. Especificação da Camada de Dados

A camada de dados foi construída de modo a satisfazer a arquitetura do sistema, os requisitos funcionais mencionados anteriormente e manter a informação consistente.

9.1. Diagramas de classes de objetos de acesso a dados

Após o diagrama de classes estar bem definido e estruturado, estipularam-se as entidades a persistir. Estas foram o *Utilizador*, *Cliente*, *Etiqueta*, *Loja*, *Receita*, *Técnica*, *Utensílios* e *Ingredientes* devido à sua importância para a consistência do sistema. Todos os dados destas classes teriam agora que ser fornecidos pela base de dados, separando assim a camada de negócio da camada de dados. Em baixo é ilustrado diagrama de classes com as entidades persistidas, ilustradas a cinzento.

**Figura 13:** Diagrama de classes para a camada de dados.

9.2. Modelação conceitual da base de dados

Foram subentendidos no diagrama de classes apresentado anteriormente, todos as entidades a serem criadas tais como os seus respectivos atributos. Dito isto foram estabelecidas as relações entre as entidades. Para uma compreensão mais legível das mesmas decidiu-se utilizar o modelo conceitual apenas para realçar estas relações, o qual é apresentado na Figura 14 e utiliza a notação Chen (Chen, 1976).

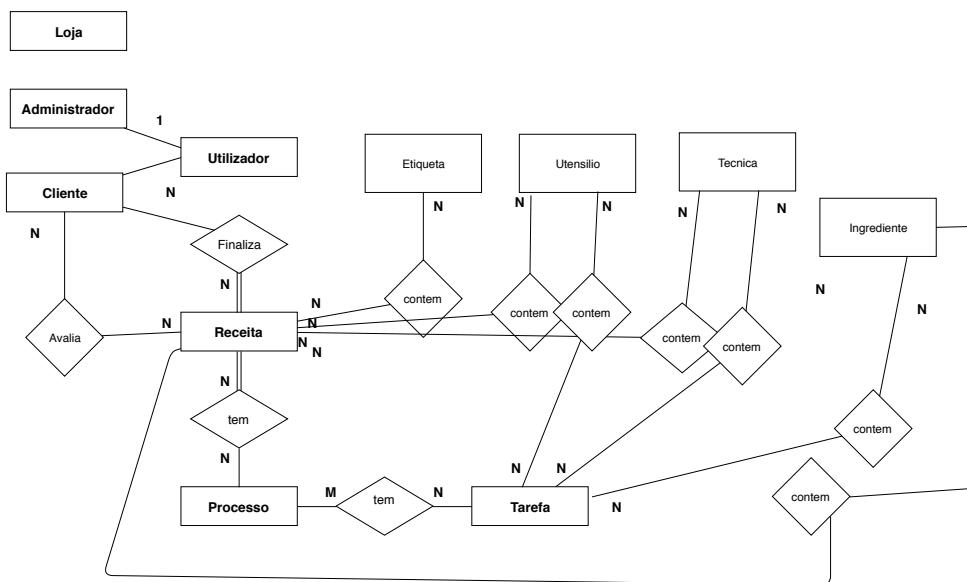


Figura 14: Modelo conceitual da base de dados.

9.3. Modelação lógica da base de dados

Tendo as relações definidas elaborou-se o modelo lógico através do programa *MySQL Workbench*¹, o qual se apresenta na Figura 15.

¹<https://www.mysql.com/products/workbench/>, acedido a 23 de maio de 2019

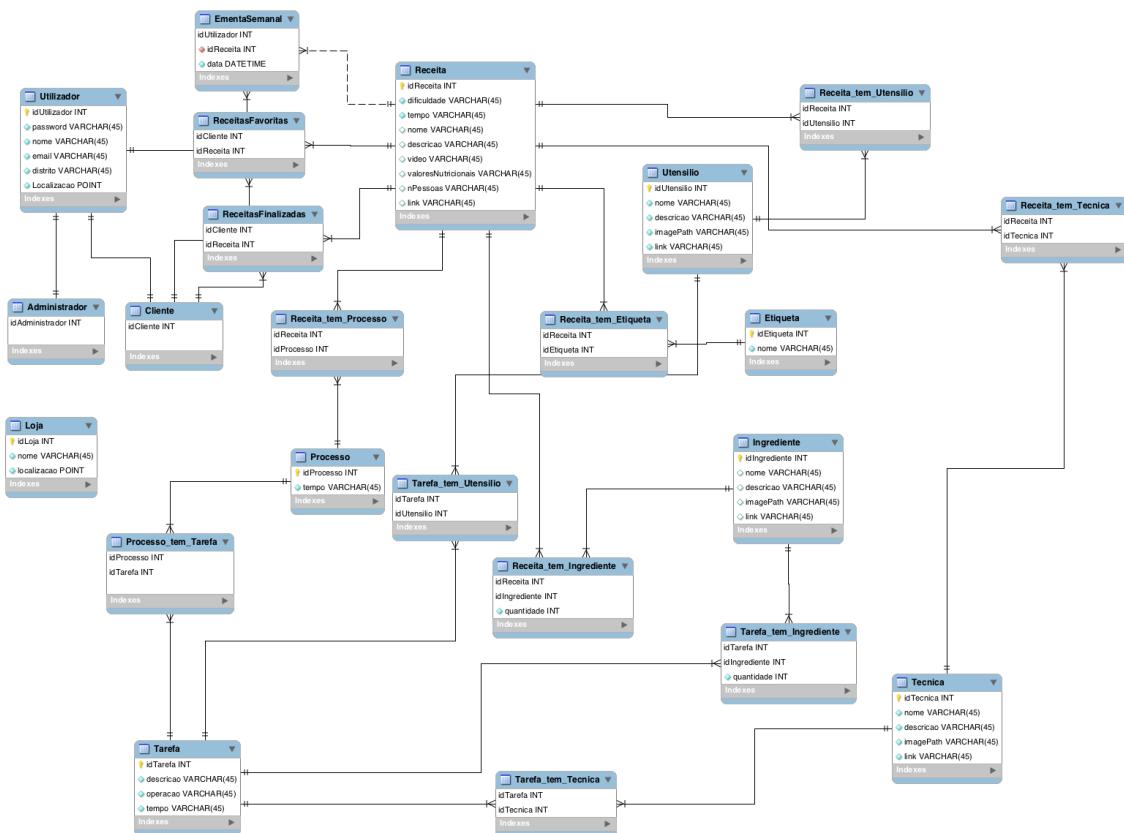


Figura 15: Modelo lógico da base de dados.

10. Especificação da Camada de Negócio

A camada de negócio engloba toda a lógica interna do sistema e as suas responsabilidades passam pelo processamento dos pedidos provenientes da camada de apresentação por forma a cumprir os requisitos funcionais do sistema, bem como assegurar o cumprimento de requisitos não-funcionais.

Assim para especificar a mesma recorreu-se a um diagrama de classes que irá apresentar as diversas classes que constituem esta camada, bem como abordar de forma mais específica todas as componentes das mesmas e diagramas de sequência que demonstram e descrevem a interação entre essas classes.

Um diagrama de pacotes onde se pode ver a estrutura genérica da camada de negócio foi já fornecido no Capítulo 8, apresentado na Figura 12, sendo o pacote referente à camada de negócio o *Model*.

10.1. Diagramas de classes

O presente diagrama de classes apresenta então como classes constituintes da camada de negócio, as classes:

- Bela Sopa;
- Utilizador;
- Administrador;
- Cliente;
- Localização;
- Ementa Semanal;
- Loja;
- Receita;
- Processo;
- Tarefa;

- Ingrediente;
- Utensílios;
- Técnicas.

As relações entre as entidades são também descritas no diagrama, fornecendo uma vista geral da estrutura do sistema.

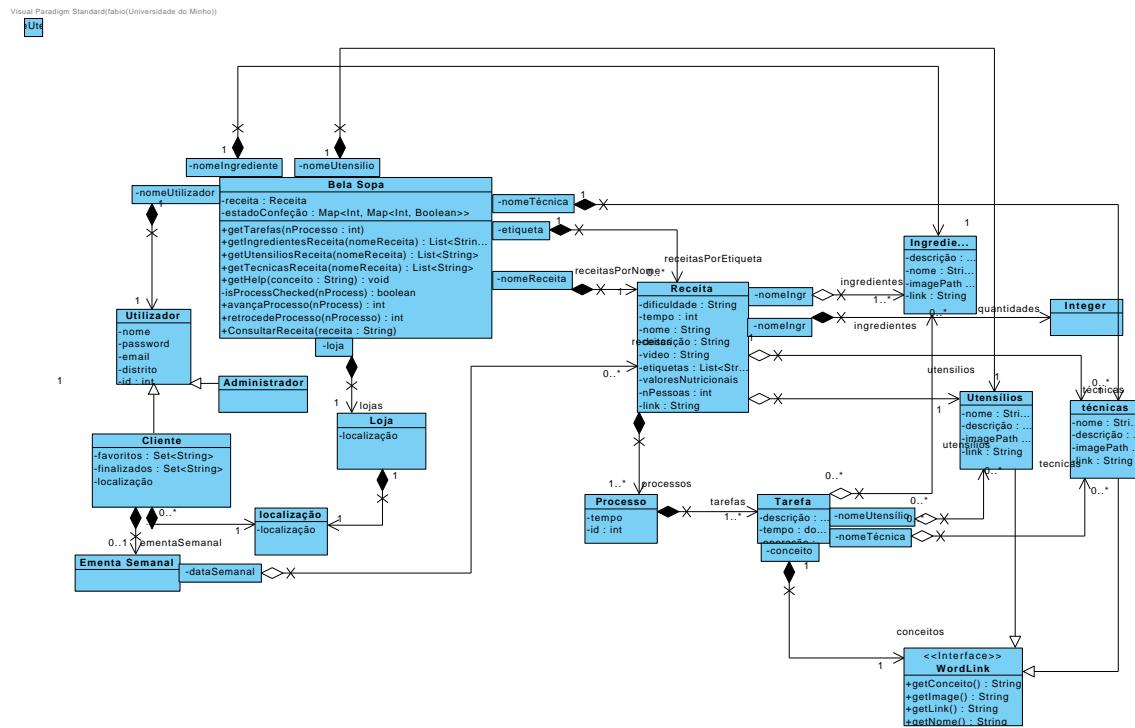


Figura 16: Diagrama de classes para a camada de negócio.

10.2. Diagramas de sequência

Para demonstrar as classes que compõem a camada de negócio, a forma como estas se relacionam e interatuam entre si, e as suas funções, apresentam-se, a título ilustrativo, os diagramas de sequência de dois eventos do sistema:

- Confeccionar Receita;
- Consultar Receita.

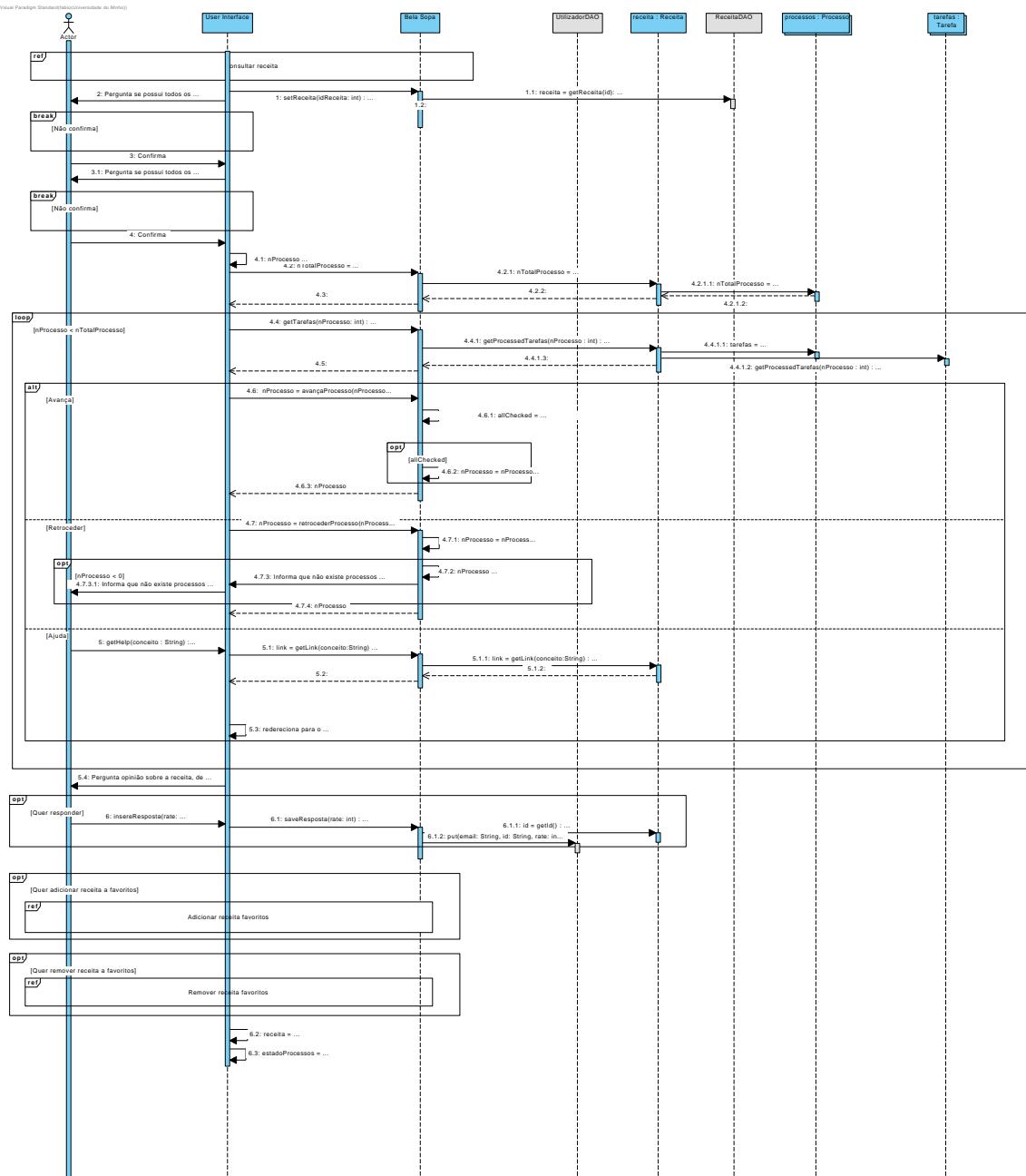


Figura 17: Diagrama de sequência para a confeção de uma receita.

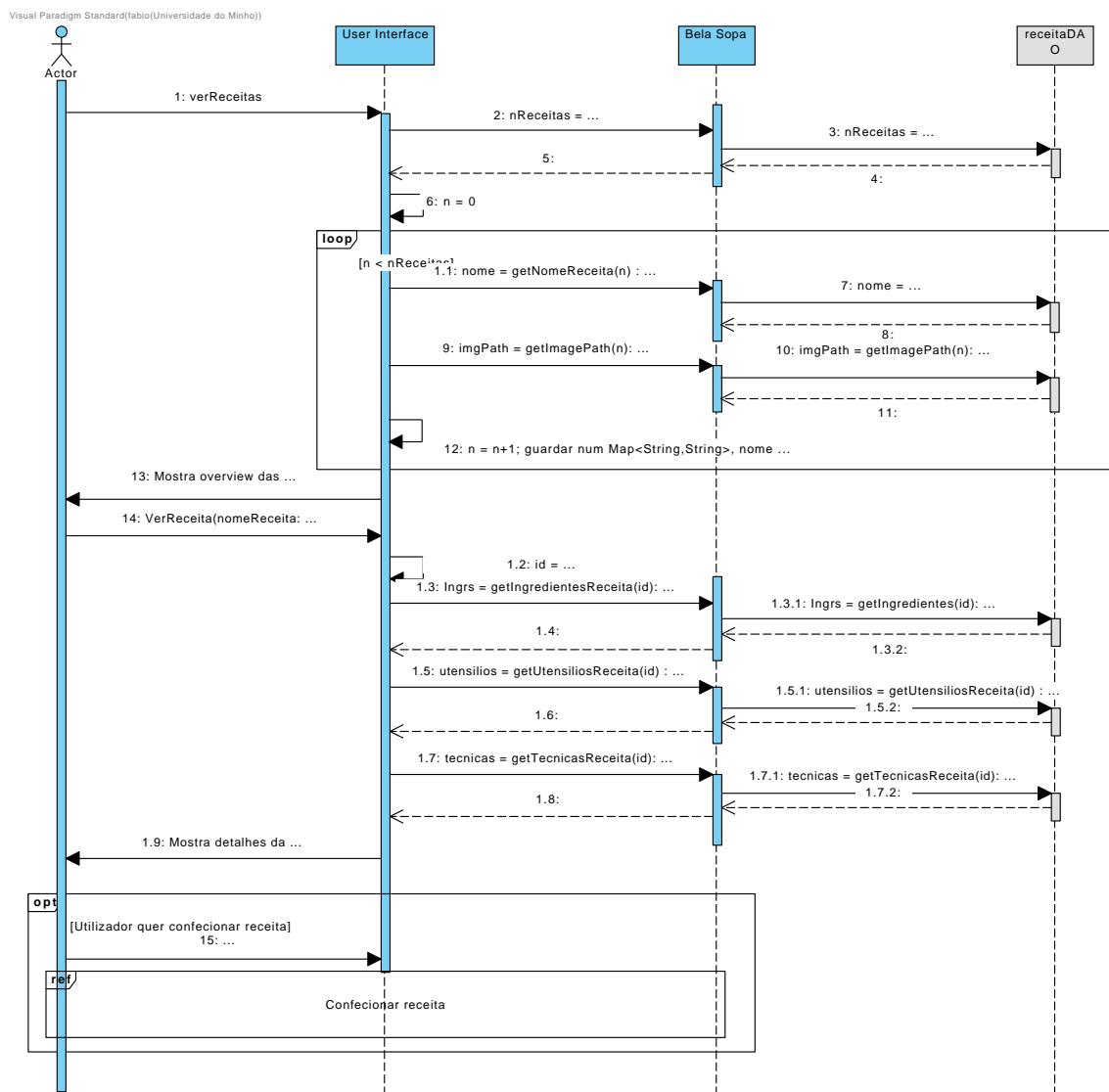


Figura 18: Diagrama de sequência para a consulta de uma receita.

Os restantes diagramas de sequência podem ser encontrados no Anexo II.

11. Construção

Tendo-se terminado o processo de especificação do sistema, procedeu-se à sua implementação. Neste capítulo são primeiramente enumeradas as principais tecnologias utilizadas na sua construção, descrevendo-se também vários aspectos relevantes da sua implementação. É depois descrito o procedimento de instalação do sistema e apresentado brevemente o produto obtido.

11.1. Tecnologias utilizadas

O sistema desenvolvido consiste em um servidor *web*, o qual é acedido por clientes remotos através de um *web browser*. Para a implementação do mesmo, utilizaram-se as seguintes principais tecnologias e *frameworks*:

- Sistema de gestão de bases de dados *Microsoft SQL Server*²;
- *Microsoft .NET Core 2.1*³, utilizando-se a linguagem C#;
- *Entity Framework Core 2.1*⁴;
- *Microsoft ASP.NET Core 2.1*⁵, utilizando-se a arquitetura *Model-View-Controller* (MVC);
- *Bing Maps API*⁶;
- *YamlDotNet 6.0*⁷.

11.2. Detalhes de implementação

O sistema desenvolvido segue o padrão arquitetural *Model-View-Controller* (MVC):

- *Models*: representam, gerem e persistem a informação utilizada pelo sistema;

²<https://www.microsoft.com/en-us/sql-server>, acedido a 23 de maio de 2019

³<https://dotnet.microsoft.com/>, acedido a 23 de maio de 2019

⁴<https://dotnet.microsoft.com/apps/aspnet>, acedido a 23 de maio de 2019

⁵<https://docs.microsoft.com/en-us/ef/core>, acedido a 23 de maio de 2019

⁶<https://www.microsoft.com/en-us/maps/choose-your-bing-maps-api>, acedido a 23 de maio de 2019

⁷<https://github.com/aaubry/YamlDotNet>, acedido a 23 de maio de 2019

- *Views*: definem a interface gráfica disponibilizada aos utilizadores do sistema, apresentando informação gerida pelos *models*;
- *Controllers*: gerem a interação do utilizador com a interface gráfica disponibilizada pelas *views*, interatuando com os *models* para obter e modificar informação.

A informação da aplicação é persistida numa base de dados relacional disponibilizada por uma instância de um servidor *Microsoft SQL Server*. O mapeamento do modelo relacional para objetos foi implementado com recurso à plataforma *Entity Framework Core*.

11.3. Procedimento de instalação

Uma vez que o sistema desenvolvido corresponde apenas a um *web server*, o processo de instalação (ou *deployment*) do mesmo envolve apenas intervenção por parte do administrador da máquina onde se deseja que este corra.

Por forma a se instalar o sistema, os seguintes pré-requisitos devem ser cumpridos:

- A máquina onde o *web server* irá correr deve utilizar alguma versão do sistema operativo *Windows* com suporte para as tecnologias enumeradas na secção anterior;
- A mesma máquina deve ter acesso (local ou remoto) a uma instância de um servido *Microsoft SQL Server*, no qual os dados da aplicação serão persistidos.

Tendo em conta que a aplicação é disponibilizada num único arquivo de ficheiros, o procedimento de instalação do sistema consiste então nos seguintes passos:

1. Extrair todo os ficheiros do arquivo da aplicação;
2. Alterar o campo `ConnectionString` do ficheiro `appsettings.json` por forma a corresponder à instância do servidor *Microsoft SQL Server* (e base de dados nesse servidor) onde os dados da aplicação devem ser persistidos;
3. Executar o programa `BelaSopa.exe`, o qual corresponde ao servidor *web* que implementa o sistema;
4. Opcionalmente, configurar o sistema operativo por forma a que este programa seja executado aquando do reinício da máquina;
5. Garantir que eventuais configurações de rede permitam que clientes remotos accedam ao servidor *web*.

Uma vez concluído este processo, os clientes terão acesso ao sistema através do seu *web browser*, especificando o endereço da máquina em que o servidor *web* está instalado.

11.4. Produto final

As Figuras 19 a 22 apresentam, a título de exemplo, o aspeto final de várias interfaces representativas disponibilizadas pelo sistema *Bela Sopa*.

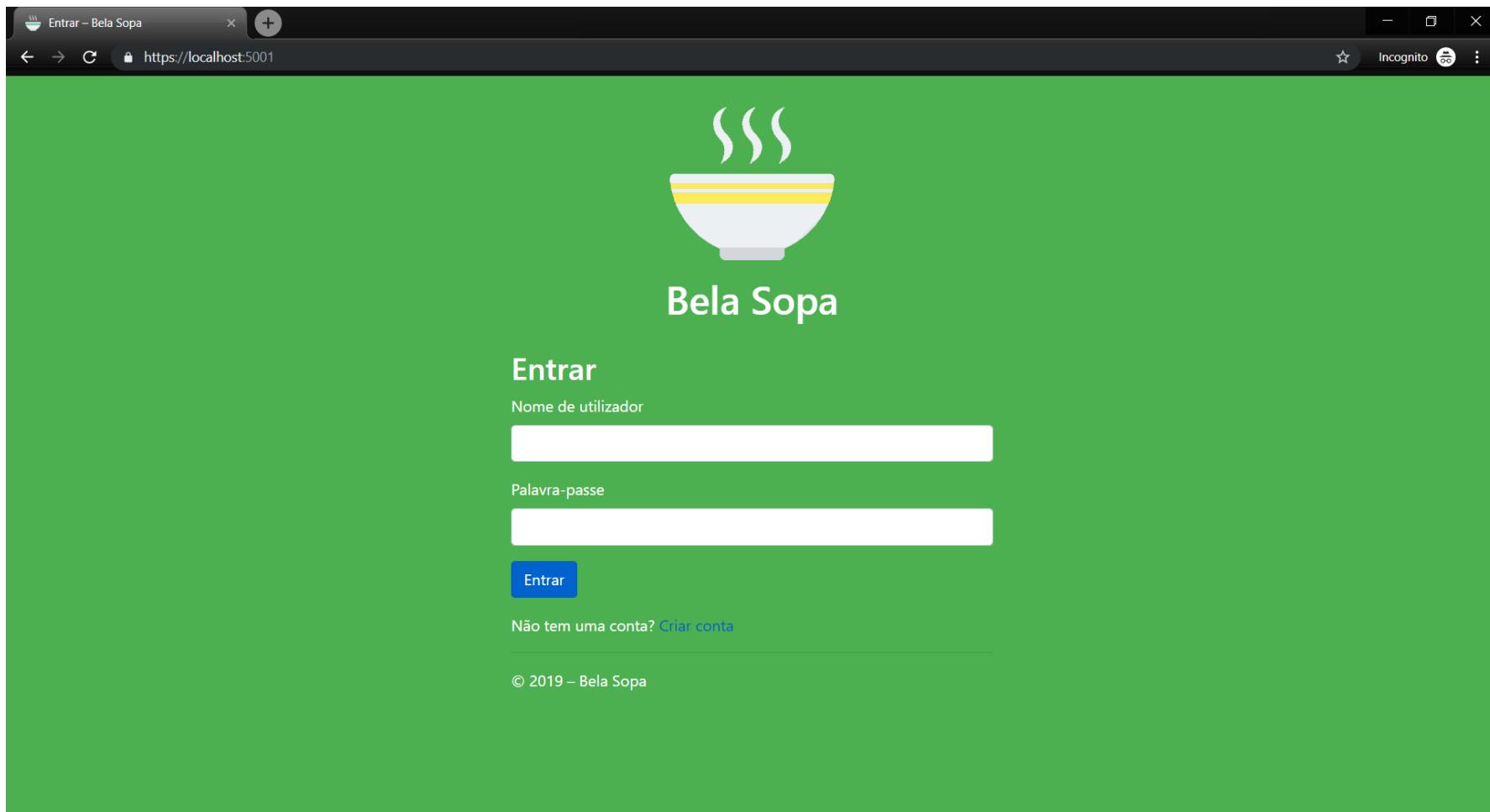


Figura 19: Aspetto final da interface de autenticação.

Receitas – Bela Sopa

https://localhost:5001/Receitas

Olá, obelo.

O meu perfil

Em confeção

Receitas

Ingredientes

Técnicas

Utensílios

Ementa semanal

Histórico

Lojas próximas

Sair

Receitas

Criar receita

Nome

Qualquer etiqueta

Qualquer dificuldade

Açorda de bacalhau com coentros e tomate seco

Arjamolho

Arjamolho com pimentos

Aveludado de abóbora com creme de

Aveludado de abóbora com iogurte

Aveludado de abóbora e tostinhos

Figura 20: Aspetto final da interface de listagem de receitas.

41

The screenshot shows a web browser window with the title 'Ingredientes – Bela Sopa'. The URL in the address bar is 'https://localhost:5001/Ingredientes'. The page has a green sidebar on the left with the 'Bela Sopa' logo and the text 'Olá, obelo.' Below the logo are several menu items: 'O meu perfil', 'Em confeção', 'Receitas', 'Ingredientes' (which is the active tab), 'Técnicas', 'Utensílios', 'Ementa semanal', 'Histórico', and 'Lojas próximas'. At the bottom of the sidebar is a 'Sair' button. The main content area is titled 'Ingredientes' and features a search bar with the placeholder 'Nome'. Below the search bar are six ingredient cards, each with an image and the name: 'Abacate' (avocado), 'Abóbora' (pumpkin), 'Agrião' (spinach), 'Aipo' (celery), 'Alface' (lettuce), and 'Alho' (garlic). The background of the main content area is white.

Figura 21: Aspetto final da interface de listagem de ingredientes.

Creme de lentilhas com abóbora x +

← → C https://localhost:5001/Receitas/Detalhes/102 Incognito ☺ :

Bela Sopa

Olá, obelo.

O meu perfil Em confeção Receitas Ingredientes Técnicas Utensílios Ementa semanal Histórico Lojas próximas Sair

← Voltar

Creme de lentilhas com abóbora ★

Média | 40 min | 8 doses

Uma sopa consistente preparada com lentilhas e abóbora. Toda a cremosidade e sabor de uma entrada perfeita, para refeição de Outono.

Etiquetas: Sabores mediterrânicos | Sem glúten | Sem leite | Vegetariana



Ingredientes para 8 doses	Quantidade
Abóbora limpa	500 g
Água	1,4 l
Azeite	2 c. de sopa
Cebola congelada picada	100 g
Cenoura rizada	300 g
Cominhos em pó	1 c. de chá
Lentilhas vermelhas	200 g
Pasta de concentrado de tomate	2 c. de sopa
Pimentão-doce	1 c. de sobremesa

Técnicas

- (nenhuma)

Utensílios

- Liquidificador
- Panela
- Varinha mágica

Figura 22: Aspetto final da interface de visualização de uma receita.

12. Conclusões e Trabalho Futuro

Neste relatório apresentou-se o trabalho prático realizado no âmbito da Unidade Curricular de Laboratórios de Informática IV, do curso de Mestrado Integrado em Engenharia Informática da Universidade do Minho, no ano letivo de 2018/2019. O caso de estudo considerado centra-se no desenvolvimento do assistente pessoal de cozinha *Bela Sopa*, encomendado pela cadeia de supermercados e hipermercados *Gota Doce*.

Foi primeiramente fundamentada a construção do sistema tendo em conta a sua utilidade e viabilidade, tendo-se também elaborado um plano para o seu desenvolvimento. Verificado-se que a construção do sistema seria vantajosa e que o seu processo de desenvolvimento cumpriria o orçamento e prazos estabelecidos, procedeu-se à fase de especificação do mesmo.

O trabalho realizado nessa mesma fase foi em seguida pormenorizado, apresentando-se com particular detalhe os resultados dos processos de modelação de domínio, levantamento e análise de requisitos, modelação de *use cases*, prototipagem da interface de utilizador e modelação da arquitetura interna do sistema.

Por fim, descreveu-se o processo de construção do sistema e apresentou-se o produto resultante, enumerando-se também as tecnologias nas quais este se baseia e especificando-se o procedimento de instalação do sistema.

Devido a restrições de tempo, algumas funcionalidades delineadas na fase de especificação do sistema não foram concretizadas aquando da sua construção. Em particular, embora as receitas confeccionadas e respetiva avaliação e dificuldade reportadas pelo utilizador sejam armazenadas e exibidas no seu histórico pessoal, esta informação não é utilizada para filtrar ou organizar a lista de receitas do sistema apresentada ao mesmo. Adicionalmente, a funcionalidade de visualização de lojas próximas poderá ser estendida com a capacidade de indicar trajetos até uma determinada loja. A implementação destas funcionalidades omissas constitui trabalho futuro.

Provando-se a adequação do sistema construído e o cumprimento dos seus objetivos, será considerado o alargamento do seu âmbito a formas de culinária além da confeção de sopas. Esta evolução é também identificada como trabalho futuro.

Referências

Chen, Peter Pin-Shan (1976). «The Entity-Relationship Model - Toward a Unified View of Data».

Em: *ACM Transactions on Database Systems* 1.1, pp. 9–36.

Object Management Group (2017). *Unified Modeling Language Specification Version 2.5.1*.

Sommerville, Ian (2010). *Software Engineering*. 9th edition. Pearson Education.

Lista de Siglas e Acrónimos

API *Application Programming Interface*

MVC *Model-View-Controller*

UML *Unified Modeling Language*

Anexo I. Especificação dos Use Cases

Incluem-se neste anexo as especificações de todos os *use cases* identificados no Capítulo 6, excetuando-se aquelas já apresentadas nesse mesmo capítulo.

Use case:	Aceder a serviços externos	
Pré-condição:	Estar autenticado	
Pós-condição:	Acedeu ao serviço externo e estado do sistema não é alterado	
	Autor	Sistema
Comportamento Normal	1. Escolhe aceder a serviços externos	
		2. Guarda o estado da aplicação
		3. Redireciona para a aplicação dos serviços externos

Tabela 5: Especificação do *use case* “aceder a serviços externos”.

Use case:	Adicionar receita aos favoritos	
Pré-condição:	Estar autenticado	
Pós-condição:	Receita foi adicionada aos favoritos	
	Autor	Sistema
Comportamento Normal	1. Escolhe adicionar a receita ao favoritos	
		2. Valida inserção
		3. Insere receita aos favoritos
		4. Informa que inseriu a receita com sucesso
Exceção 1 [Receita já está nos favoritos] (passo 3)		3.1. Informa que a receita está nos favoritos

Tabela 6: Especificação do *use case* “adicionar receita aos favoritos”.

Use case:	Apagar conta	
Pré-condição:	Estar autenticado	
Pós-condição:	Conta eliminada	
	Autor	Sistema
Comportamento Normal	1. Fornece o email da conta que pretende ser apaga.	
		2. Valida email
		3. Elimina conta
Comportamento Alternativo 1 [Autor é cliente] Exceção 1 [Email de cliente inválido] (passo 2)		2.1. Valida se a conta é do próprio
		2.2. Elimina conta
Exceção 2 [Dados inválidos] (passo 2)		2.1. Informa que o email é inválido
Exceção 2 [Dados inválidos] (passo 2)		2.1. Informa que o email é invalido

Tabela 7: Especificação do *use case* “apagar conta”.

Use case:	Consultar ou editar informação da conta	
Pré-condição:	Estar autenticado	
Pós-condição:	Dados consultados	
	Autor	Sistema
Comportamento Normal	1. Consulta perfil	
		2. Mostra o perfil do autor
	3. Conclui Consulta	
Comportamento Alternativo 1 [Autor edita perfil] (passo 3)	3.1. Edita perfil	
		3.2. «include» Apagar Conta
		3.3. «include» Registar Conta

Tabela 8: Especificação do *use case* “consultar ou editar informação da conta”.

Use case:	Consultar Receita	
Pré-condição:	Estar autenticado, receita existe	
Pós-condição:	Receita foi consultada	
	Autor	Sistema
Comportamento Normal	1. Escolhe receita que deseja consultar	
		2. Mostra a descrição, as imagens, a porção, os utensílios, ,os ingredientes, etc...
	3. «extends» Confeccionar receita	

Tabela 9: Especificação do *use case* “consultar receita”.

<i>Use case:</i>	Editar ementa semanal	
Pré-condição:	Estar autenticado	
Pós-condição:	Ementa semanal foi alterada	
	Autor	Sistema
Comportamento Normal		1. Mostra todas as receitas na ementa nos dias e refeições já escolhidas 2. «extends» adicionar receita à ementa
Comportamento Alternativo 1 [Escolhe remover uma receita] (passo 2)		2.1. «extends» remover receita da ementa

Tabela 10: Especificação do *use case* “editar ementa semanal”.

<i>Use case:</i>	Gerar lista de ingredientes	
Pré-condição:	Estar autenticado	
Pós-condição:	Lista de ingredientes guardada no sistema	
	Autor	Sistema
Comportamento Normal	1. Decide gerar a lista de ingredientes 2. Consulta ementa semanal do autor 3. Calcula os ingredientes necessários para a semana 4. Mostra os ingredientes e as respetivas quantidades 5. Confirma a lista de ingredientes. 6. Guarda a lista de ingredientes 7. «extends» aceder a serviços externos	
Comportamento Alternativo 1 [Autor quer adicionar ingrediente à lista] (passo 5)	5.1. Escolhe o ingrediente e quantidade a adicionar 5.2. Adiciona ingrediente à lista 5.3. Volta ao passo 5	
Comportamento Alternativo 2 [Autor quer remover ingredientes à lista] (passo 5)	5.1. Escolhe o ingrediente e quantidade a remover 5.2. Remove ingrediente da lista 5.3. Volta ao passo 5	
Comportamento Alternativo 3 [Quantidade do ingrediente é zero] (passo 5.2)		5.2.1 Indica que o ingrediente não existe na lista 5.2.2 Volta ao passo 5

Tabela 11: Especificação do *use case* “gerar lista de ingredientes”.

<i>Use case:</i>	Pedir ajuda	
Pré-condição:	Estar autenticado	
Pós-condição:		
	Autor	Sistema
Comportamento Normal	1. Solicita ajuda num certo conceito	
		2. Procura conceito
		3. Mostra uma descrição, e possivelmente uma imagem e/ou video sobre o conceito
Exceção 1 [Não existe conceito] (passo 2)		2.1. Informa que não é possível encontrar esse conceito

Tabela 12: Especificação do *use case* “pedir ajuda”.

<i>Use case:</i>	Remover receita dos favoritos	
Pré-condição:	Estar autenticado	
Pós-condição:	Receita foi removida aos favoritos	
	Autor	Sistema
Comportamento Normal	1. Escolhe remover a receita dos favoritos	
		2. Valida remoção
		3. Remove receita dos favoritos
		4. Informa que removeu a receita com sucesso
Exceção 1 [Receita não se encontra nos favoritos] (passo 3)		3.1. Informa que a receita não está nos favoritos

Tabela 13: Especificação do *use case* “remover receita dos favoritos”.

<i>Use case:</i>	Ver loja mais próxima	
Pré-condição:	Estar autenticado e existem lojas	
Pós-condição:	foi encontrada uma loja	
	Autor	Sistema
Comportamento Normal	1. Solicita ver a loja mais perto de uma determinada localização	
		2. Calcula a loja mais perto das conhecidas
		3. Mostra loja

Tabela 14: Especificação do *use case* “ver loja mais próxima”.

Anexo II. Diagramas de Sequência para a Camada de Negócio

Apresenta-se neste anexo os restantes diagramas de sequência que descrever o funcionamento, dos métodos que envolvem interação entre as componentes/classes da camada de negócio.

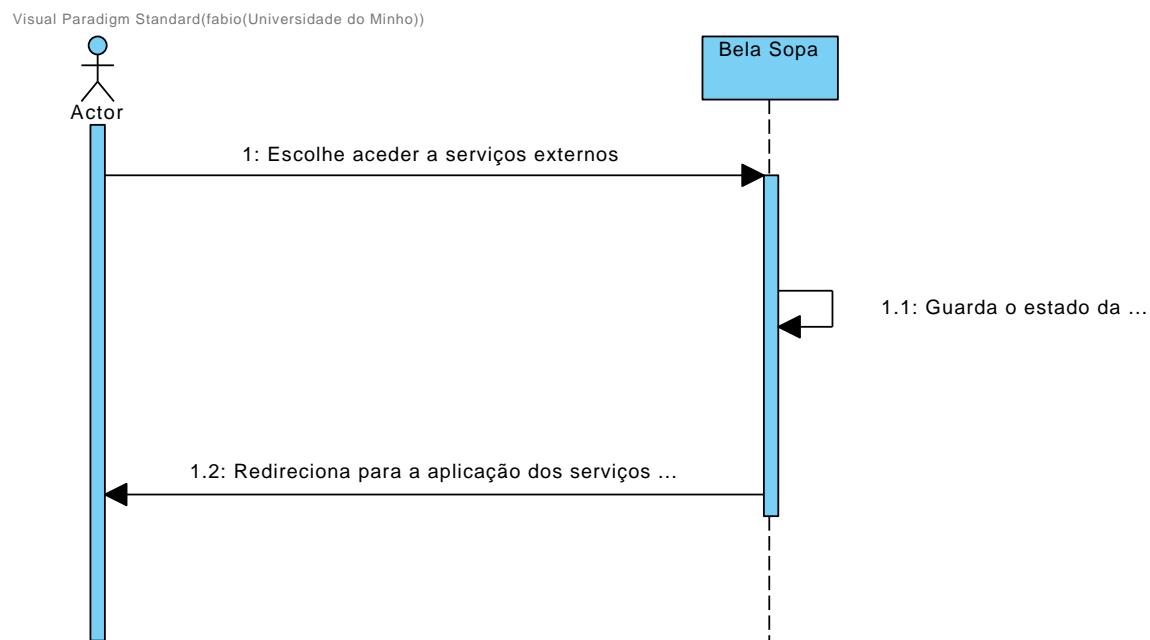
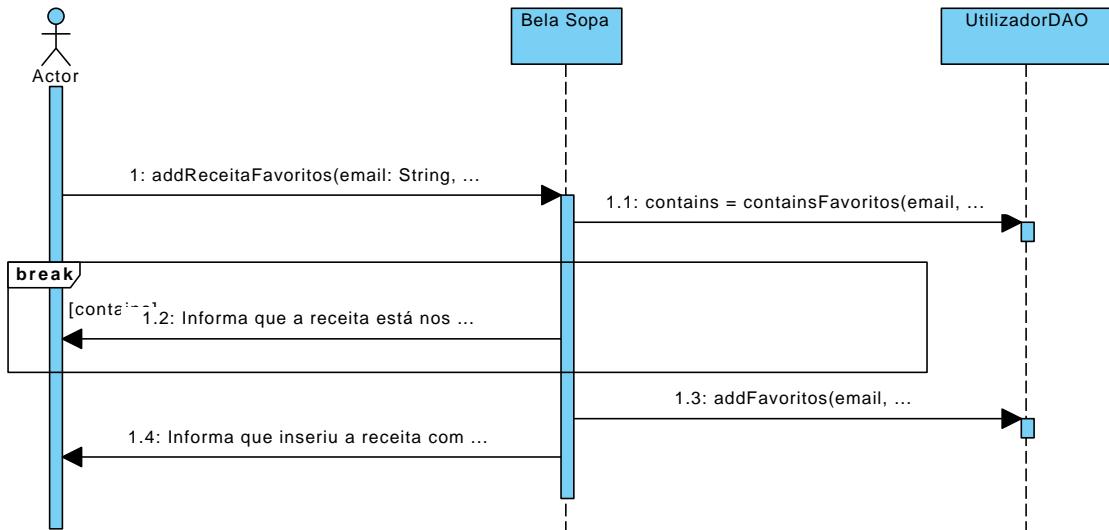
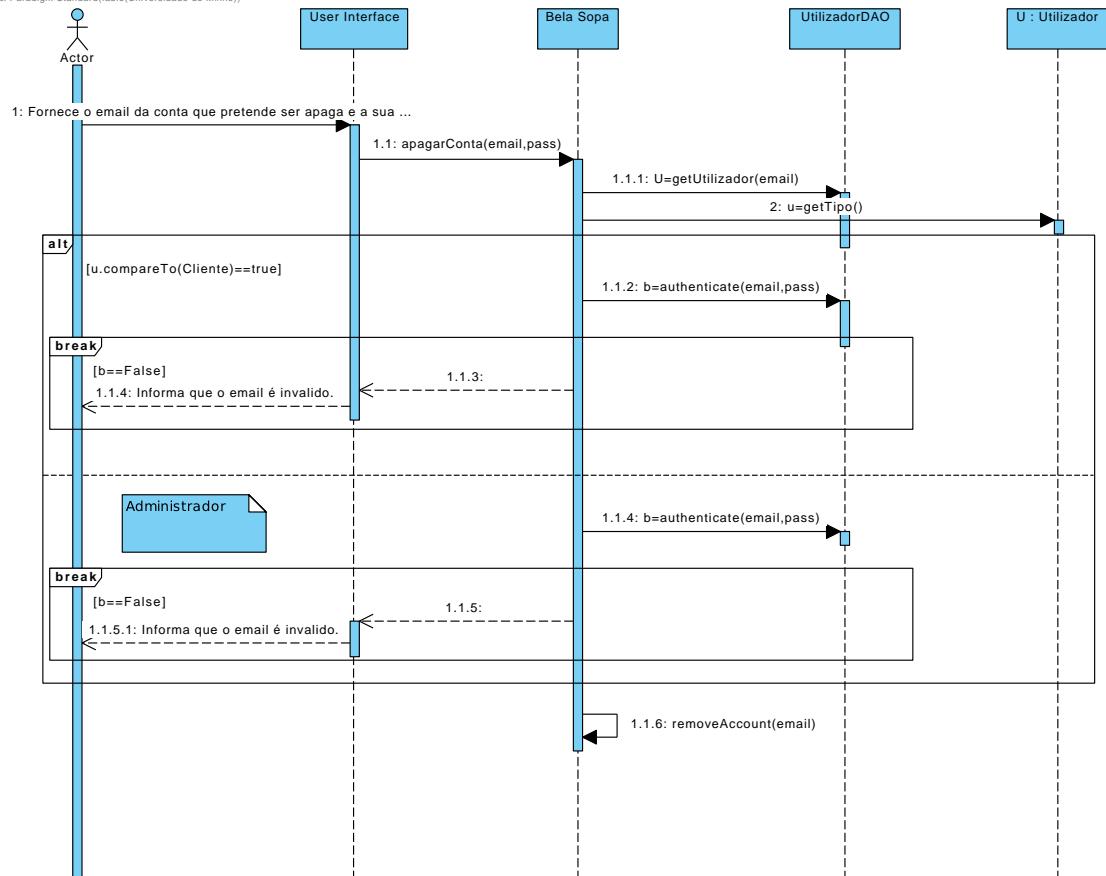


Figura 23: Diagrama de sequência para o acesso a serviços externos.

**Figura 24:** Diagrama de sequência para o adicionar de uma receita aos favoritos.**Figura 25:** Diagrama de sequência para a remoção de uma conta.

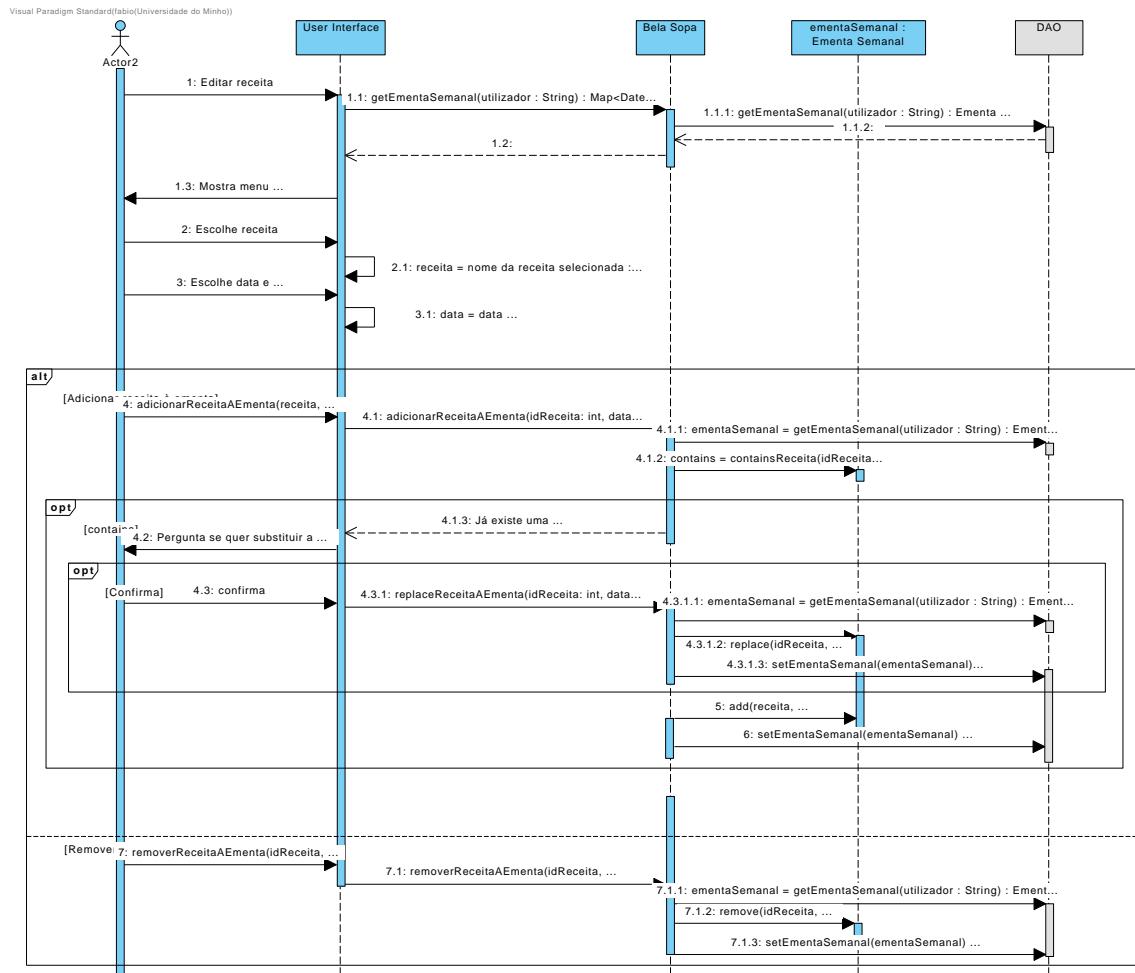


Figura 26: Diagrama de sequência para a edição da ementa semanal.

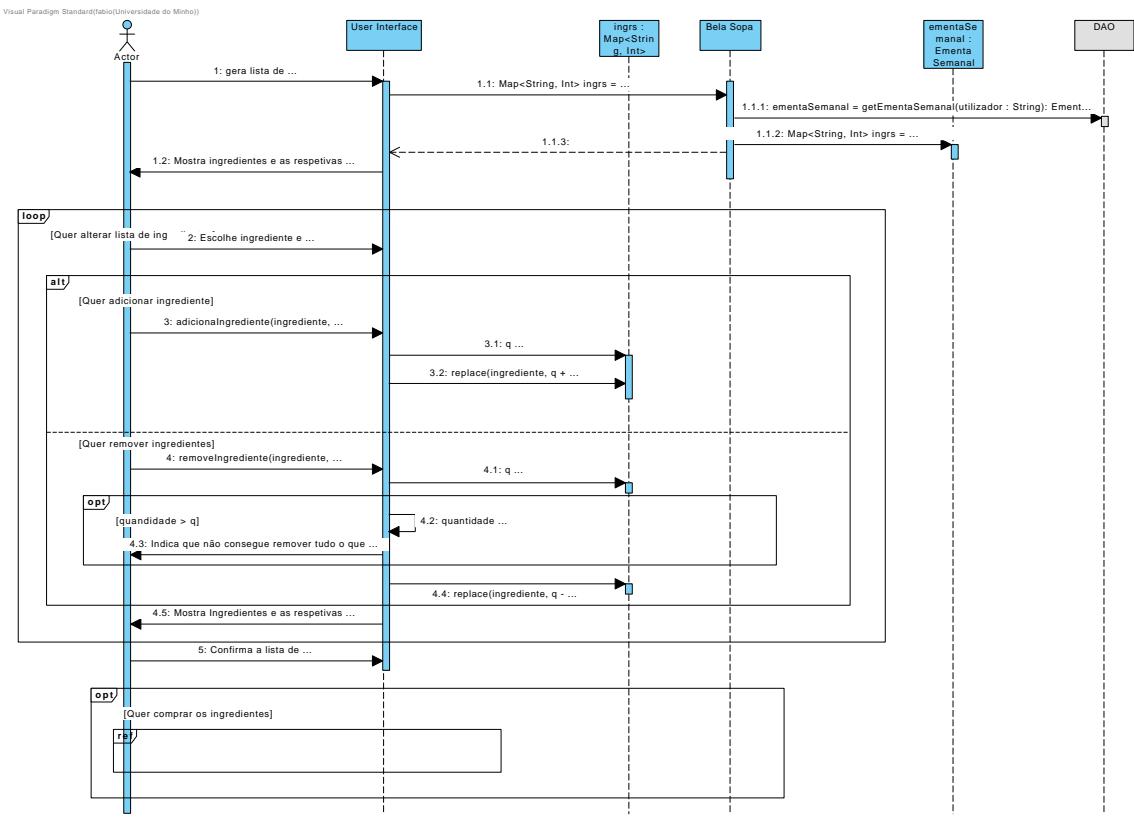


Figura 27: Diagrama de sequência para a geração de uma lista de ingredientes.

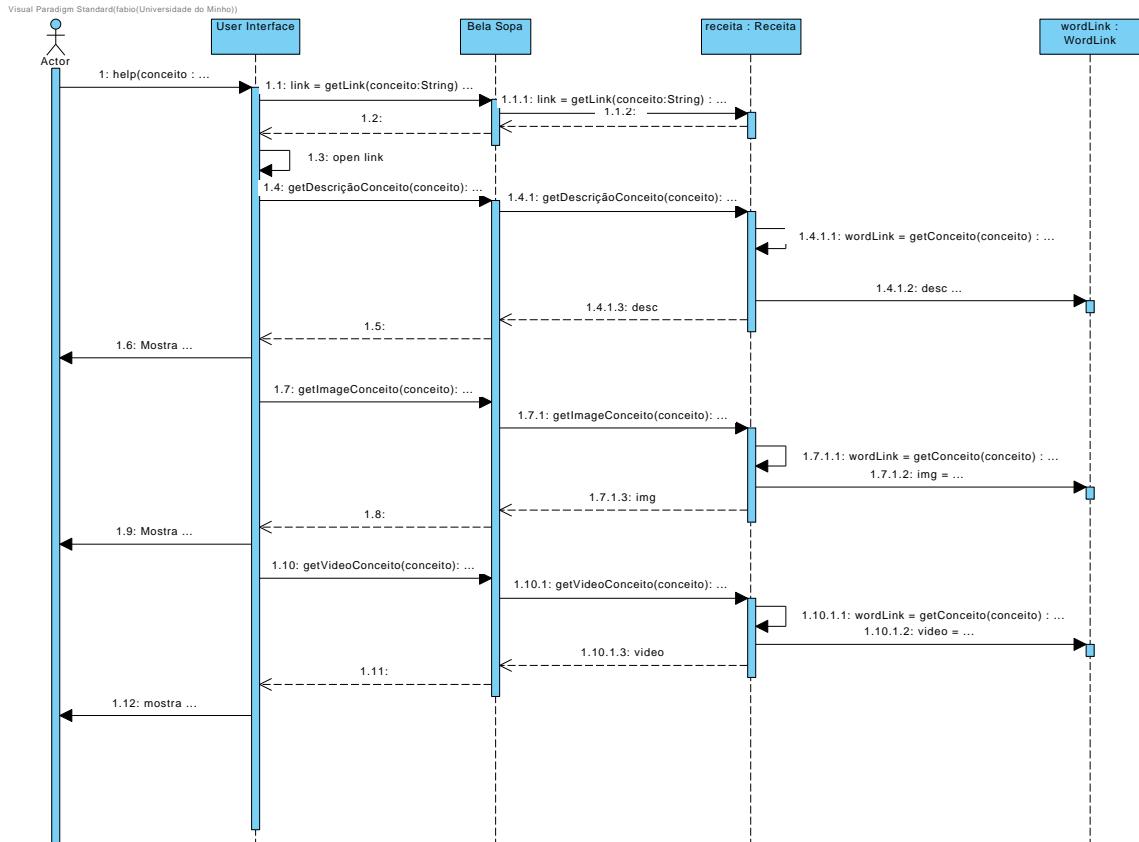


Figura 28: Diagrama de sequência para o efetuar de um pedido de ajuda.

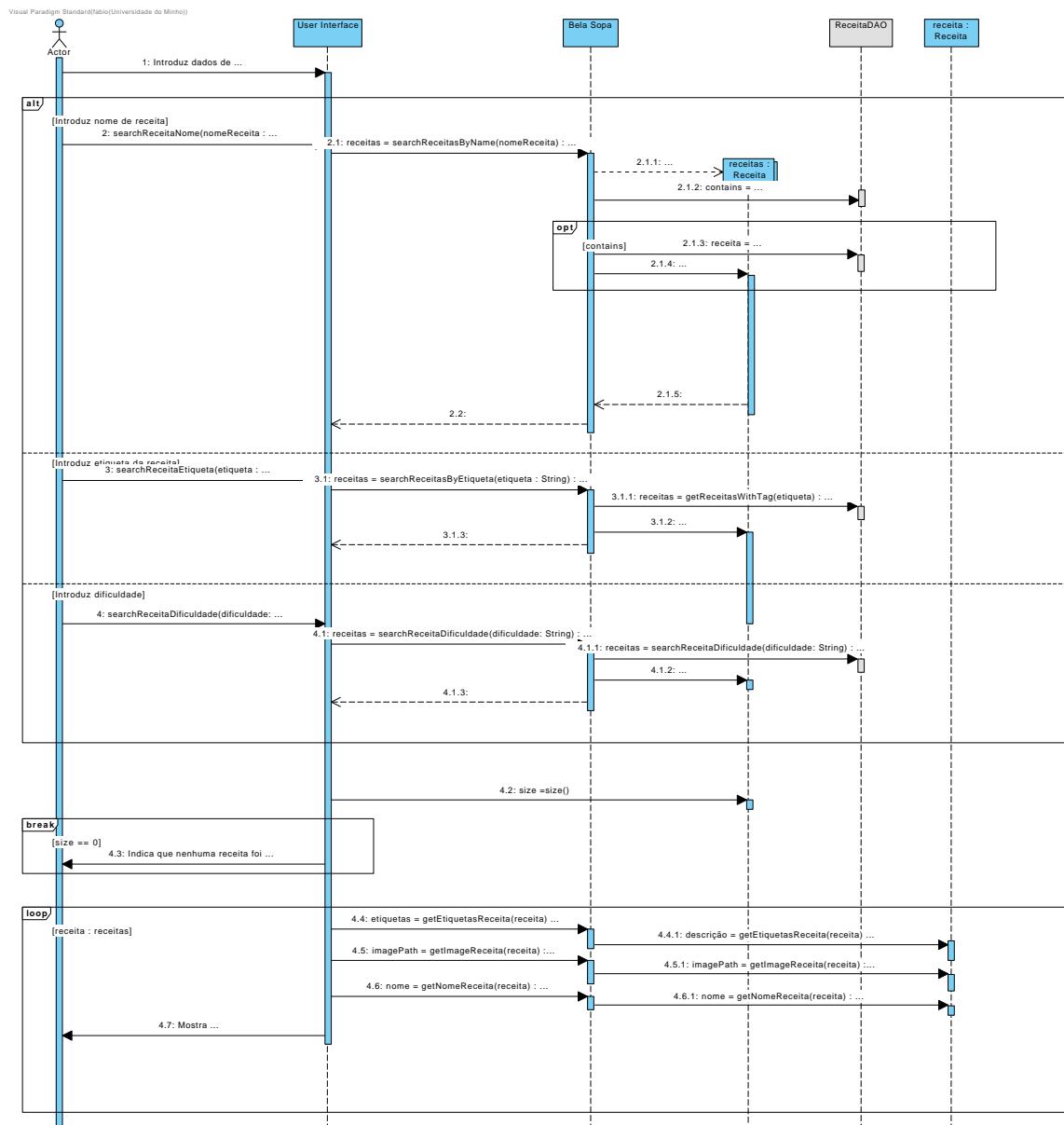


Figura 29: Diagrama de sequência para a procura de uma receita.

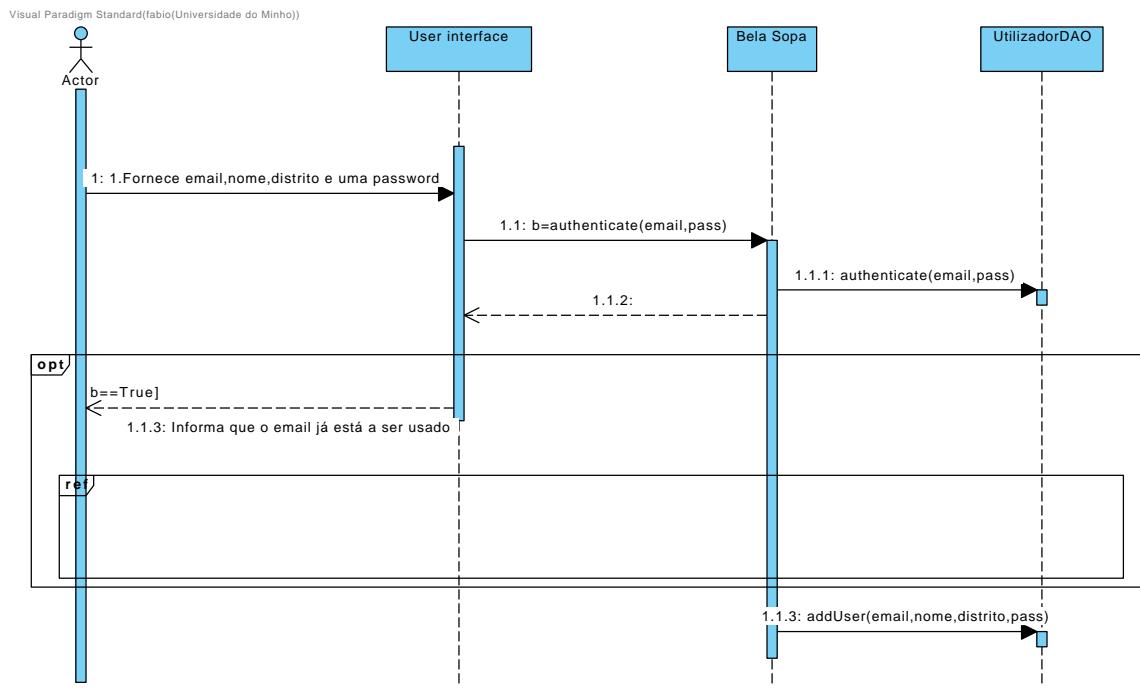


Figura 30: Diagrama de sequência para a criação de uma conta.

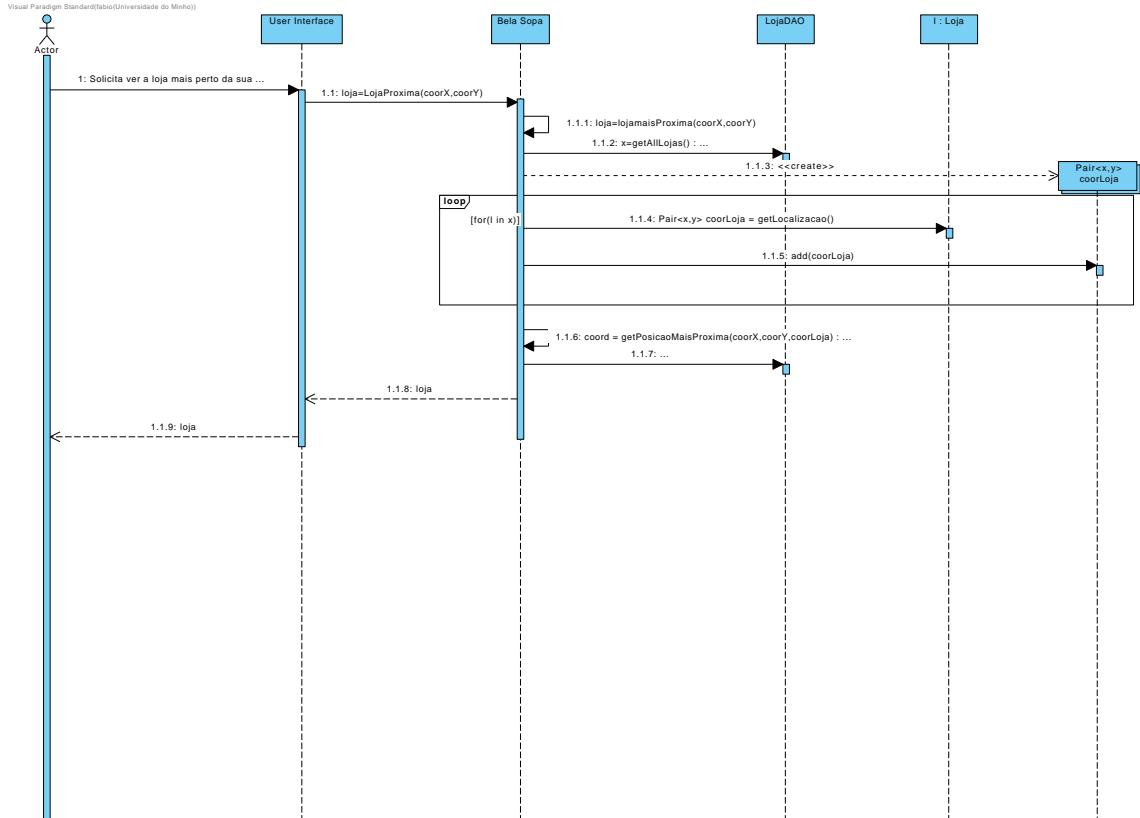


Figura 31: Diagrama de sequência para a visualização de lojas próximas.

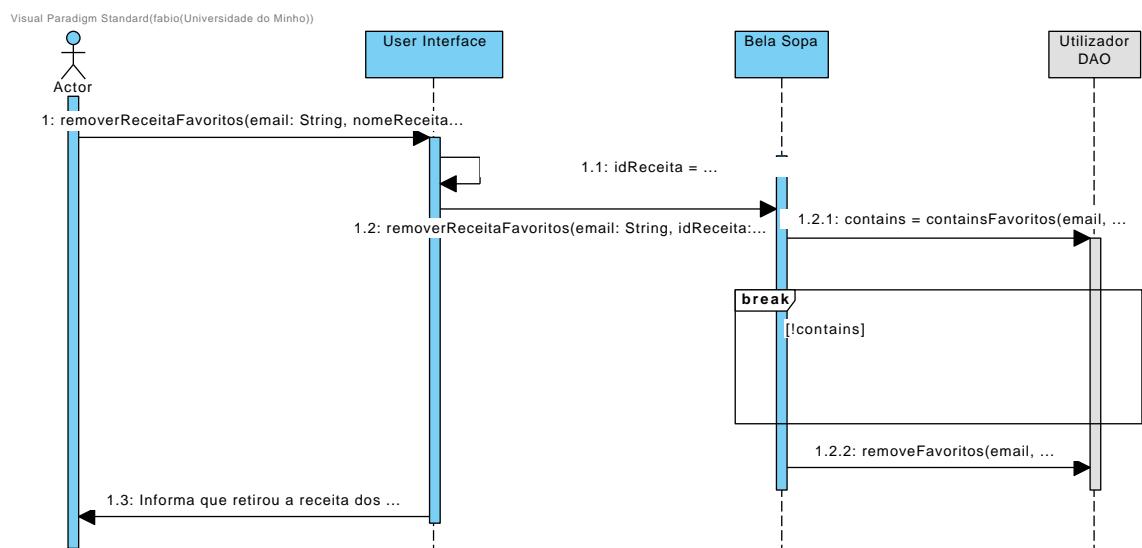


Figura 32: Diagrama de sequência para a remoção de uma receita dos favoritos.